Software Reference for x510 Series Switches

AlliedWare Plus™ Operating System Version 5.4.4-3.5

AT-x510-28GTX
AT-x510-28GPX
AT-x510-52GTX
AT-x510-52GPX
AT-x510-28GSX (built-in AC PSUs)
AT-x510-28GSX-80 (built-in DC PSUs)
AT-x510DP-28GTX
AT-x510DP-52GTX
AT-x510L-28GT
AT-x510L-28GP
AT-x510L-52GT
AT-x510L-52GP

C613-50053-01 REV A

Allied Telesis

Acknowledgments

This product includes software developed by the University of California, Berkeley and its contributors.

Copyright ©1982, 1986, 1990, 1991, 1993 The Regents of the University of California.

All rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. For information about this see **www.openssl.org/**

Copyright ©1998-2008 The OpenSSL Project. All rights reserved.

This product includes software licensed under the GNU General Public License available from: **www.gnu.org/licenses/gpl2.html**

Source code for all GPL licensed software in this product can be obtained from the Allied Telesis GPL Code Download Center at: **www.alliedtelesis.com/support/default.aspx**

Allied Telesis is committed to meeting the requirements of the open source licenses including the GNU General Public License (GPL) and will make all required source code available.

If you would like a copy of the GPL source code contained in Allied Telesis products, please send us a request by registered mail including a check for US$15 to cover production and shipping costs and a CD with the GPL code will be mailed to you.

GPL Code Request
Allied Telesis Labs (Ltd)
PO Box 8011
Christchurch
New Zealand

©2014 Allied Telesis Inc. All rights reserved.

This documentation is subject to change without notice. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's internal use without the written permission of Allied Telesis, Inc.

Allied Telesis, AlliedWare Plus, EPSRing, SwitchBlade, and VCStack are trademarks or registered trademarks in the United States and elsewhere of Allied Telesis, Inc. Adobe, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Additional brands, names and products mentioned herein may be trademarks of their respective companies.

Getting the most from this manual

Although you can view this document using Acrobat version 5, to get the best from this manual, we recommend using Adobe Acrobat Reader version 8 or later. You can download Acrobat free from **www.adobe.com/**

New features in this software version

For information about new and enhanced features in this version, see **Appendix B: Changes in this Software Version** or **New and Enhanced Features in AlliedWare Plus 5.4.4 Major and Minor Versions**. This document can be downloaded from the Support area of our website at **www.alliedtelesis.com**. Note that to download software files, you need a valid user account.

Table of Contents

Part 1: Setting up the Switch

Chapter 1: Getting Started

Introduction 1.2
How to Login 1.2
How to Get Command Help 1.3
  Viewing a List of Valid Parameters 1.3
  Completing Keywords 1.5
  Viewing Command Error Messages 1.6
How to Work with Command Modes 1.7
  Entering Privileged Exec Commands when in a Configuration Mode 1.9
How to See the Current Configuration 1.10
Default Settings 1.11
The Default Configuration 1.12
How to Change the Password 1.13
How to Set Strong Passwords 1.14
How to Set an IP Address on VLAN 1 ..... 1.16
How to Save and Boot from the Current Configuration 1.17
  How to Save to the Default Configuration File 1.17
  How to Create and Use a New Configuration File 1.17
How to Return to the Factory Defaults 1.19
How to See System Information 1.20
  Viewing Overall System Information 1.20
  Viewing Voltage, Fan Status, and Temperature 1.20
  Viewing the Serial Number 1.21
How to Set System Parameters 1.21
  How to Change the Telnet Session Timeout 1.21
  How to Name the Switch 1.23
  How to Display a Text Banner at Login 1.24
How to Set the Time and Date 1.25
  How to Show Current Settings 1.25
  How to Set the Time and Date 1.25
  How to Set the Timezone 1.26
  How to Configure Summer-Time 1.26
How to Add and Remove Users 1.27
  Pre-Encrypted Passwords 1.28
How to Undo Settings 1.30
  How to Use the *no* Parameter 1.30
  How to Use the *default* Parameter 1.30
How to Upgrade the Firmware 1.31
Save Power with the Eco-Friendly Feature 1.32
  The Eco-Friendly LED Feature 1.32
  The Eco-Friendly LPI Feature 1.32
Trouble-shoot Fiber and Pluggable Issues 1.33
Using the Find Me feature 1.34
Continuous Reboot Prevention 1.35

Controlling “show” Command Output 1.37
AlliedWare Plus GUI 1.39

Chapter 2: Command Syntax Conventions in this Software Reference

Chapter 3: Start-Up Sequence

AlliedWare Plus Start-Up 3.2
Diagnostic Menu 3.3
Bootloader Menu 3.4
Start-Up Sequence 3.9

Chapter 4: CLI Navigation Commands

Command List 4.2

Chapter 5: User Access Commands

Introduction 5.2
Command List 5.2
- Using plain passwords 5.5
- Using encrypted passwords 5.5
- Using hidden passwords 5.6
- Using plain passwords 5.8
- Using encrypted passwords 5.8
- Using hidden passwords 5.9

Chapter 6: Creating and Managing Files

Introduction 6.2
USB Support 6.2
Working with Files 6.2
- Listing Files 6.2
- Displaying the Contents of Configuration and Text Files 6.4
- Navigating Through the File System 6.4
- Using the Editor 6.6

Creating and Using Configuration Files 6.8
- Creating a Configuration File 6.8
- Specifying the Start-Up Configuration Script 6.8
- Working with Configuration Files 6.9
- The Configuration File Fallback Order 6.10

Copying Files to and from Your Device 6.12
- URL Syntax 6.12
- Copying Files 6.13
- Copying from a Server to Running Configuration 6.17

The Autoboot Feature 6.18
- Restoring a Switch Using Autoboot from External Media 6.19
- Configure Autoboot 6.21

Chapter 7: File Management Commands

Introduction 7.3
- URL Syntax and Keyword Usage 7.3

Command List 7.5

Chapter 8: Licensing Introduction and Configuration

Introduction 8.2

About Licensing VCStack 8.2
Licensing Terminology 8.2
Applying a New Feature License on a Stand-Alone Switch 8.4
VCStack Licensing Configuration 8.5
Adding a New Switch to a VCStack 8.5
Adding a New Feature License to a VCStack 8.6

Chapter 9: Licensing Commands

Command List 9.2

Chapter 10: System Configuration and Monitoring Commands

Command List 10.2

Chapter 11: Debugging and Logging

Introduction 11.2
Debugging 11.2
Logging to a Terminal 11.2
Turning Off Debugging 11.3
Logging 11.3
Log Outputs 11.4

Chapter 12: Logging Commands

Command List 12.2

Chapter 13: Scripting Commands

Command List 13.2

Chapter 14: Interface Commands

Command List 14.2

Chapter 15: Interface Testing Commands

Command List 15.2

Part 2: Layer Two Switching

Chapter 16: Switching Introduction

Introduction 16.2
Physical Layer Information 16.3
Switch Ports 16.3
Activating and Deactivating Switch Ports 16.4
Autonegotiation 16.4
Duplex Mode 16.4
Speed Options 16.4
MDI/MDIX Connection Modes 16.5
The Layer 2 Switching Process 16.7
The Ingress Rules 16.7
The Learning Process 16.8
The Forwarding Process 16.9
The Egress Rules 16.9

Layer 2 Filtering 16.11
Ingress Filtering 16.11
Storm Control 16.12
Loop Protection 16.13
Loop Detection 16.13
Thrash Limiting 16.14
Support for Jumbo Frames 16.16
Port Mirroring 16.17
Port Security 16.18
MAC Address Learn Limits 16.18
IEEE 802.1X 16.18
Static MAC Address 16.18
Quality of Service 16.19
IGMP Snooping 16.20

Chapter 17: Switching Commands

Command List 17.2

Chapter 18: VLAN Introduction

VLANs Introduction 18.2
Virtual LANs (VLANs) 18.2
Configuring VLANs 18.3
VLAN Double Tagging (VLAN Stacking) 18.5
How Double-Tagged VLANs Work 18.5
VLAN Rules for Double Tagging 18.5
Restrictions when Using Double-Tagged VLANs 18.6
Configuring Double-Tagged VLANs 18.6
Private VLANs 18.11
Private VLANs for Ports in Access Mode 18.11
Private VLAN Operation with Ports in Access Mode 18.13
Access Mode Private VLAN Configuration Example 18.14
Private VLANs for Trunked Ports 18.17
Trunked Port Private VLAN Configuration Example 18.18
Protocol Based VLAN Configuration Example 18.22
VLAN Statistics 18.25
Counter Operation 18.25

Chapter 19: VLAN Commands

Command List 19.2

Chapter 20: Spanning Tree Introduction: STP, RSTP, and MSTP

Introduction 20.2
Overview of Spanning Trees 20.2
Spanning Tree Operation 20.2
Spanning Tree Modes 20.4
Spanning Tree Protocol (STP) 20.5
Configuring STP 20.6
Rapid Spanning Tree Protocol (RSTP) 20.8
Configuring RSTP 20.9
Multiple Spanning Tree Protocol (MSTP) 20.11
Multiple Spanning Tree Instances (MSTI) 20.12
MSTP Regions 20.13
Common and Internal Spanning Tree (CIST) 20.15
MSTP Bridge Protocol Data Units (BPDUs) 20.17

Configuring MSTP 20.19

Chapter 21: Spanning Tree Commands

Command List 21.3

Chapter 22: Link Aggregation Introduction and Configuration

Introduction and Overview 22.2
Static and Dynamic (LACP) Link Aggregation 22.3
- Static Channel Groups 22.3
- Dynamic (LACP) Channel Groups 22.3
- Link Aggregation Control Protocol (LACP) 22.3

Configuring an LACP Channel Group 22.5
- Minimal LACP Group Configuration 22.8

Configuring a Static Channel Group 22.9

Chapter 23: Link Aggregation Commands

Introduction 23.2
Command List 23.3

Chapter 24: Power over Ethernet Introduction

Introduction 24.2
- PoE Standards 24.2
- PoE (all standards) 24.3
- PoE (IEEE 802.3af) 24.3
- Enhanced PoE 24.3
- PoE+ (IEEE 802.3at) 24.3
- Differences Between PoE and PoE+ 24.5
- LLDP-MED (TIA-1057) with PoE+ (IEEE 802.3at) 24.5
- PoE and PoE+ Applications 24.5
- Power Device (PD) Discovery 24.5
- Power Classes 24.6
- Power through the Cable 24.7
- Cable Types 24.8
- Static and Automatic Power Allocation 24.8

PoE and PoE+ Implementation 24.9
- Power Capacity 24.9
- PoE Port Allocation and Distribution 24.9
- Power Threshold 24.10
- Negotiating Power Requirements 24.10
- PoE Port Management 24.10
- Powered Device (PD) Detection 24.11
- Port Prioritization 24.11
- Software Monitoring 24.13

PoE and PoE+ Configuration 24.14
- Add a Description for a PoE or PoE+ Port 24.14
- Configuring Capacity and Priority on a PoE or PoE+ Port 24.15
- Remotely Monitoring Power for all Connected PDs 24.16

Chapter 25: Power over Ethernet Commands

Introduction 25.2
Command List 25.2

Chapter 26: GVRP Introduction and Configuration

Introduction 26.2
GVRP Example 26.3
GVRP Guidelines 26.4
GVRP and Network Security 26.5
GVRP-Inactive Intermediate Switches 26.5
Enabling GVRP on the Switch 26.5
Enabling GVRP on the Ports 26.6
Setting the GVRP Timers 26.6
Disabling GVRP on the Ports 26.7
Disabling GVRP on the Switch 26.7
Configuring and Validating GVRP 26.8

Chapter 27: GVRP Commands

Command List 27.2

Part 3: Layer Three Switching and Routing

Chapter 28: Internet Protocol (IP) Addressing and Protocols

Introduction 28.2
Address Resolution Protocol (ARP) 28.3
Static ARP Entries 28.3
Timing Out ARP Entries 28.3
Deleting ARP Entries 28.4
Proxy ARP 28.4
ARP Logging 28.7
Domain Name System (DNS) 28.8
Domain Name Parts 28.8
Server Hierarchy 28.8
DNS Client 28.9
DNS Relay 28.11
DHCP Options 28.12
Internet Control Message Protocol (ICMP) 28.13
Checking IP Connections 28.14
Ping 28.14
Traceroute 28.14
IP Helper 28.15
IP Directed Broadcast 28.16

Chapter 29: IP Addressing and Protocol Commands

Introduction 29.2
Command List 29.3

Chapter 30: IPv6 Introduction

Introduction 30.2
Overview 30.2
IPv6 Addresses and Prefixes 30.3
Address Types 30.3
IPv6 Headers 30.5
The Internet Control Message Protocol (ICMPv6) 30.7
IPv6 Routing 30.10
Integration of IPv4 and IPv6 ..... 30.10

IPv6 on your Switch 30.11
Enabling IPv6 ..... 30.11
IPv6 Stateless Address Autoconfiguration (SLAAC) 30.11
IPv6 EUI-64 Addressing 30.11
IPv6 Link-local Addresses 30.12
IPv6 RA Guard 30.13
RA Guard Introduction 30.13
Enabling IPv6 RA Guard 30.13

Chapter 31: IPv6 Commands

Command List 31.2

Chapter 32: IPv6to4 Tunneling Configuration

Introduction 32.2
6to4 Automatic Tunnel Configuration 32.2
Tunneling Operation 32.2
6to4 Tunnels, Operation and Configuration Summary 32.3
Example 6to4 Configuration 32.4

Chapter 33: IPv6to4 Tunneling Commands

Command List 33.2

Chapter 34: Routing Protocol Overview

Introduction 34.2
RIP 34.2
OSPF 34.2
VRRP 34.3

Chapter 35: Route Selection

Introduction 35.2
Types of Routes 35.2
Interface Routes 35.2
Static Routes 35.2
Dynamic Routes 35.3
RIB and FIB Routing Tables 35.4
Understanding the Routing Information Base (RIB) 35.4
Administrative Distance 35.6
Metric 35.8
Equal Cost Multipath Routing 35.8
How AlliedWare Plus Deletes Routes 35.8
How AlliedWare Plus Adds Routes 35.9
Troubleshooting Routes not Installed in the RIB 35.10
Troubleshooting Routes not Installed in the FIB 35.10

Chapter 36: Routing Commands

Introduction 36.2
Command List 36.3

Chapter 37: RIP Configuration

Introduction 37.2
Enabling RIP 37.2

Specifying the RIP Version 37.4
RIPv2 Authentication (Single Key) 37.6
RIPv2 Text Authentication (Multiple Keys) 37.8
RIPv2 md5 authentication (Multiple Keys) 37.12

Chapter 38: RIP Commands

Introduction 38.2
Command List 38.2

Chapter 39: RIPng for IPv6 Configuration

Introduction 39.2
Enabling RIPng 39.2
Troubleshooting RIPng Adjacency 39.5

Chapter 40: RIPng for IPv6 Commands

Introduction 40.2
Command List 40.2

Chapter 41: OSPF Introduction and Configuration

OSPF Introduction 41.2
Features 41.2
OSPF Components 41.2
- Autonomous Systems 41.2
- Routing Areas 41.3
- Adjacencies and Designated Routers 41.3
- Link State Advertisements 41.4
- OSPF Packet Types 41.4
- OSPF States 41.5
- OSPF Metrics 41.6
- Automatic Cost Calculation 41.7
- Routing with OSPF 41.7
- Network Types 41.8
- Passive Interfaces 41.8
- Authenticating OSPF 41.9
- Redistributing External Routes 41.9

Enabling OSPF on an Interface 41.10
Setting Priority 41.12
Configuring an Area Border Router 41.14
OSPF Cost 41.15
Configuring Virtual Links 41.18
OSPF Authentication 41.20
OSPF Multi-Area Loopback Configuration 41.23

Chapter 42: OSPF Commands

Introduction 42.3
Command List 42.3

Chapter 43: OSPFv3 for IPv6 Introduction and Configuration

OSPFv3 Introduction 43.2
Features 43.2
Licensing 43.3
Routing Overview 43.3

OSPF Components 43.4
Autonomous Systems 43.4
Routing Areas 43.4
Relationships Between Routers 43.4
OSPFv3 Packet Types 43.4
Link State Advertisements (LSAs) 43.8
LSA Header 43.9
OSPFv3 States 43.10
OSPFv3 Metrics 43.10
Automatic Cost Calculation 43.11
Network Types 43.11
Passive Interfaces 43.12
Redistributing External Routes 43.12
Differences Between OSPFv2 and OSPFv3 ..... 43.13
Protocol Processing Applied Per-Link Rather than Per-Subnet 43.13
Removed Address Semantics 43.13
Neighbors are Identified by Router ID 43.13
New Link-Local Flooding Scope for Link State Advertisements 43.14
Uses Link-Local Unicast Addresses 43.14
Support Provided for Multiple OSPF Instances per Link 43.14
Unknown LSAs are Handled More Effectively 43.14
OSPFv3 Authentication and Encryption Overview 43.15
OSPFv3 Authentication and Encryption Support 43.16
OSPFv3 Virtual Links 43.17
Configuring OSPFv3 ..... 43.18
Example 1: Single-Area Network Configuration 43.19
Example 2: Two-Area Network Configuration 43.26
Setting Up the Metrics 43.28
Configuring OSPFv3 Authentication and Encryption 43.33
Configuring OSPFv3 Authentication on a VLAN 43.34
Configuring OSPFv3 Encryption on a VLAN 43.35
Configuring OSPFv3 Authentication in an OSPFv3 Area 43.36
Configuring OSPFv3 Encryption in an OSPFv3 Area 43.37
Configuring OSPFv3 Authentication and Encryption for a Virtual Link 43.38
OSPFv3 Authentication in an OSPFv3 Area 43.40
OSPFv3 Encryption in an OSPFv3 Area 43.42
OSPFv3 Authentication on a VLAN 43.44
OSPFv3 Encryption on a VLAN 43.46
OSPFv3 Authentication with Two VLANs 43.48
OSPFv3 Encryption with Two VLANs 43.50
OSPFv3 Authentication for a Virtual Link 43.52
OSPFv3 Encryption for a Virtual Link 43.54

Chapter 44: OSPFv3 for IPv6 Commands

Command List 44.3

Chapter 45: Route Map Configuration

Introduction 45.2
Structure of a Route Map 45.2
Clauses 45.2
Configuring Route Maps for Filtering and Modifying OSPF Routes 45.3
Configuring a Match Clause 45.3
Configuring a Set Clause 45.4
Applying Route Maps in OSPF 45.5

Chapter 46: Route Map Commands

Command List 46.2

Part 4: Multicast Applications

Chapter 47: Multicast Introduction and Commands

Multicast Introduction 47.2
Multicast Groups 47.3
Components in a Multicast Network 47.3
Command List 47.5

Chapter 48: IGMP and IGMP Snooping Introduction

Introduction 48.2
IGMP 48.2
Joining a Multicast Group (Membership Report) 48.3
Staying in the Multicast Group (Query Message) 48.4
Leaving the Multicast Group (Leave Message) 48.4
IGMP Snooping 48.5
How IGMP Snooping Operates 48.5
IGMP Snooping and Querier Configuration Example 48.6
Query Solicitation 48.9
How Query Solicitation Works 48.9
Query Solicitation Operation 48.9
Speeding up IGMP Convergence in a Non-Looped Topology 48.12
Enabling Query Solicitation on Multiple Switches in a Looped Topology 48.12

Chapter 49: IGMP and IGMP Snooping Commands

Introduction 49.2
Command List 49.2

Chapter 50: PIM-SM Introduction and Configuration

Introduction 50.2
PIM-SM 50.2
Characteristics of PIM-SM 50.2
Roles in PIM-SM 50.3
Operation of PIM-SM 50.4
PIM-SM Configuration 50.6
Static Rendezvous Point configuration 50.7
Dynamic Rendezvous Point Configuration 50.9
Bootstrap Router Configuration 50.10
PIM-SSM 50.13
Characteristics of PIM-SSM 50.13
PIM-SSM IP Address Range 50.13
IGMPv3 and SSM-Mapping 50.13
How PIM-SSM Works 50.14
How IGMP SSM-Mapping Works 50.15
Configure PIM-SSM 50.15

Chapter 51: PIM-SM Commands

Command List 51.2

Chapter 52: PIM-SMv6 Introduction and Configuration

Introduction 52.2
PIM-SMv6 ..... 52.2
Characteristics of PIM-SMv6 ..... 52.3
Roles in PIM-SMv6 ..... 52.4
PIM-SMv6 Terminology 52.5
Operation of PIM-SMv6 ..... 52.6
Data Flow from Source to Receivers for PIM-SMv6 ..... 52.8
PIM-SMv6 Embedded RP, RP and BSR Candidate Configurations 52.10
Embedded RP Configuration 52.10
Verify Embedded RP Configuration 52.11
RP and BSR Candidate Configuration 52.13
Verify RP and RP Candidate Configuration 52.15
PIM-SMv6 Static RP, DR, BSR Configurations 52.17
Static Rendezvous Point Configuration 52.18
Verify Static Rendezvous Point Configuration 52.19
Dynamic Rendezvous Point Configuration 52.20
Verify PIM Group-to-RP Mappings 52.21
Verify RP Details 52.22
Boot Strap Router Configuration 52.22
Verify Boot Strap Router Configuration 52.23

Chapter 53: PIM-SMv6 Commands

Command List 53.2

Chapter 54: PIM-DM Introduction and Configuration

Introduction 54.2
Characteristics of PIM-DM 54.2
PIM-DM Terminology 54.3
PIM-DM Configuration 54.4
Configuration Example 54.4
Verifying Configuration 54.6

Chapter 55: PIM-DM Commands

Command List 55.2

Chapter 56: MLD and MLD Snooping Introduction and Commands

MLD Introduction 56.2
MLD Snooping 56.3
MLD Snooping Configuration Examples 56.4
Command List 56.7

Part 5: Access and Security

Chapter 57: Access Control Lists Introduction

Introduction 57.2
Overview 57.2
ACL Rules 57.3
ACL Source and Destination Addresses 57.3
ACL Reverse Masking 57.3
Hardware and Software ACL Types 57.4

Defining Hardware MAC ACLs 57.5
Defining Hardware IP ACLs 57.6
Actions for Hardware ACLs 57.7
Attaching Hardware ACLs to Interfaces 57.7
Hardware ACLs and QoS Classifications 57.8
Classifying Your Traffic 57.8
Security ACLs 57.8
QoS ACLs 57.9
Attaching Hardware ACLs Using QoS 57.9
Filtering Hardware ACLs with QoS 57.11
Using QoS Match Commands with TCP Flags 57.11
ACL Filter Sequence Numbers 57.14
ACL Filter Sequence Number Behavior 57.14
ACL Filter Sequence Number Applicability 57.14
ACL Filter Sequence Number Types 57.15
ACL Filter Sequence Configuration 57.18
Creating ACLs in Global Configuration Mode 57.20
Display the ACL configuration details 57.22

Chapter 58: IPv4 Hardware Access Control List (ACL) Commands

Introduction 58.2
IPv4 Hardware Access List Commands and Prompts 58.3
Command List 58.4

Chapter 59: IPv4 Software Access Control List (ACL) Commands

Introduction 59.2
IPv4 Software Access List Commands and Prompts 59.3
Command List 59.4

Chapter 60: IPv6 Hardware Access Control List (ACL) Commands

Introduction 60.2
IPv6 Hardware Access List Commands and Prompts 60.3
Command List 60.4

Chapter 61: IPv6 Software Access Control List (ACL) Commands

Introduction 61.2
IPv6 Software Access List Commands and Prompts 61.3
Command List 61.4

Chapter 62: Quality of Service (QoS) Introduction

Introduction 62.2
QoS Operations 62.2
QoS Packet Information 62.3
Link Layer QoS 62.3
Differentiated Services Architecture 62.4
The Differential Services Field 62.5
Processing Pre-Marked Packets 62.6
Applying QoS on Your Switch 62.7
Classifying Your Data 62.7
Class Maps 62.7
Policy Maps 62.10
Premarking and Remarking Your Traffic 62.11
CoS to Egress Queue Premarking 62.11

DSCP to Egress Queue Premarking 62.13
Rate Limiting Traffic 62.14
Policing (Metering) Your Data 62.14
Single-Rate Three-Color Policing 62.15
Two-Rate Three-Color Policing 62.16
Configuring and Applying a Policer 62.17
Remarking Your Data 62.18
Configuring the Egress Queues 62.19
Egress Queues and QoS markers 62.19
Egress Queue Commands Hierarchy 62.19
Egress Queue Shaping 62.21
Scheduling 62.21
Drop Mode 62.22
Storm Protection 62.24
Policy-Based Routing 62.25
Practical Example 62.25

Chapter 63: QoS Commands

Command List 63.2

Chapter 64: Authentication Introduction and Configuration

Authentication Introduction 64.2
Configuring a Guest VLAN 64.2
802.1X-Authentication 64.3
Web-Authentication 64.4
What is Web-Authentication? 64.4
Web-Authentication Basics 64.4
Configuring Web-Authentication 64.6
Starting a Web-Authentication Session 64.8
Configuring an External Login Page 64.10
Customizing Web-Authentication Login Page 64.13
Support for Protocols Underlying Web-Authentication 64.15
Web-Authentication Timeout Connect 64.20
Web Authorization Proxy 64.20
MAC-Authentication 64.21
Why is MAC-Authentication Required? 64.21
How Does MAC-Authentication Work? 64.21
Configuring MAC-Authentication 64.22
Tri-Authentication 64.23
Tri-Authentication Configuration 64.24
Two-Step Authentication 64.25
Ensuring Authentication Methods Require Different Usernames and Passwords 64.26
Roaming Authentication 64.27
Roaming Authentication Overview 64.28
Roaming Authentication Feature Interactions 64.29
Unauthenticated Supplicant Traffic 64.30
Deciding When a Supplicant Fails Authentication 64.32
Failed Authentication VLAN 64.33
Limitations on Allowed Feature Combinations 64.33

Chapter 65: Authentication Commands

Command List 65.3

Chapter 66: AAA Introduction and Configuration

AAA Introduction 66.2
Available Functions and Server Types 66.2
Server Groups and Method Lists 66.3
Configuring AAA Login Authentication 66.5
AAA Configuration Tasks 66.5
Sample Authentication Configurations 66.7
Sample 802.1X-Authentication Configuration 66.7
Sample MAC-Authentication Configuration 66.8
Sample Web-Authentication Configuration 66.9
Sample Tri-Authentication Configuration 66.10

Chapter 67: AAA Commands

Command List 67.2

Chapter 68: 802.1X Introduction and Configuration

Introduction 68.2
802.1X System Components 68.2
The 802.1X Implementation 68.5
Configuring 802.1X 68.7

Chapter 69: 802.1X Commands

Command List 69.2

Chapter 70: RADIUS Introduction and Configuration

Introduction 70.2
RADIUS Packets 70.3
RADIUS Attributes 70.4
RADIUS Security 70.5
RADIUS Proxy 70.6
RADIUS Accounting 70.7
RADIUS Configuration 70.9
Switch Configuration Tasks 70.9
Switch to RADIUS Server Communication 70.10
AAA Server Groups Configuration 70.12
RADIUS Configuration Examples 70.15
RADIUS Authentication 70.15
Single RADIUS Server Configuration 70.16
Multiple RADIUS Server Configuration 70.17
RADIUS Server Group Configuration 70.17
RADIUS Server Configuration using Server Groups 70.18

Chapter 71: RADIUS Commands

Command List 71.2

Chapter 72: Local RADIUS Server Introduction and Configuration

Local RADIUS Server Introduction 72.2
Enable the Local RADIUS Server 72.2
Add the Local RADIUS Server as a RADIUS Server 72.3
Add Authenticators to the List of Authenticators 72.3
Configure the Local RADIUS Server User Database 72.4
Authenticating Login Sessions 72.5

RADIUS Authentication with User Privileges 72.5
Creating Certificates for Single Users and All Users 72.8
Defined RADIUS Attributes List 72.9

Chapter 73: Local RADIUS Server Commands

Command List 73.2

Chapter 74: TACACS+ Introduction and Configuration

Introduction 74.2
TACACS+ Overview 74.2
The AlliedWare Plus TACACS+ Implementation 74.2
Authentication 74.3
Authorization 74.3
Accounting 74.4
Configuration 74.5
Configure TACACS+ 74.5
TACACS+ Configuration Example 74.7

Chapter 75: TACACS+ Commands

Command List 75.2

Chapter 76: Secure Shell (SSH) Introduction

Introduction 76.2
Secure Shell on the AlliedWare Plus OS 76.2
Configuring the SSH Server 76.4
Creating a Host Key 76.4
Enabling the Server 76.4
Modifying the Server 76.5
Validating the Server Configuration 76.6
Adding SSH Users 76.6
Authenticating SSH Users 76.7
Adding a Login Banner 76.7
Monitoring the Server and Managing Sessions 76.8
Debugging the Server 76.8
Configuring the SSH Client 76.9
Modifying the Client 76.9
Adding SSH Servers 76.10
Authenticating with a Server 76.10
Connecting to a Server and Running Commands 76.11
Copying files to and from the Server 76.11
Debugging the Client 76.11

Chapter 77: Secure Shell (SSH) Configuration

SSH Server Configuration Example 77.2

Chapter 78: Secure Shell (SSH) Commands

Introduction 78.2
Command List 78.2

Chapter 79: DHCP Snooping Introduction and Configuration

Introduction 79.2

DHCP Snooping 79.2
DHCP Snooping Database 79.3
DHCP Relay Agent Option 82 ..... 79.4
Traffic Filtering with DHCP Snooping 79.6
ARP Security 79.8
MAC Address Verification 79.8
DHCP Snooping Violations 79.8
Interactions with Other Features 79.9
Configuration 79.10
Configure DHCP Snooping 79.10
Disabling DHCP Snooping 79.16
Related Features 79.16

Chapter 80: DHCP Snooping Commands

Command List 80.2

Part 6: Network Availability

Chapter 81: VRRP Introduction and Configuration

VRRP Introduction 81.2
Virtual Router Redundancy Protocol 81.3
VRRP Configuration for IPv4 ..... 81.4
VRRP Election and Preempt for IPv4 ..... 81.6
VRRP Configuration for IPv6 ..... 81.8
VRRP Election and Preempt for IPv6 ..... 81.10
VRRP Debugging 81.12
VRRP Configuration Examples 81.13
VRRP Preferred Master with Backup Configuration 81.13
VRRP Circuit Failover Configuration 81.16
VRRPv2 to VRRPv3 Transition Configuration 81.23
VRRP IPv6 Configuration Example 81.29

Chapter 82: VRRP Commands

Command List 82.2

Chapter 83: EPSR Introduction and Configuration

Introduction 83.2
Ring Components and Operation 83.2
Fault Detection and Recovery 83.4
Fault Recovery 83.4
Restoring Normal Operation 83.5
Managing Rings with Two Breaks 83.6
Recovery When One Break is Restored 83.8
Configuration Examples 83.10
Single Domain, Single Ring Network 83.10
Single Ring, Dual Domain Network 83.16
Interconnected Rings 83.17
Superloop Protection 83.18
EPSR Superloop Prevention 83.19
Configuring a Basic Superloop Protected Two Ring EPSR Network 83.22
Sample Show Output 83.38
Adding a New Data VLAN to a Functioning Superloop Topology 83.42

EPSR and Spanning Tree Operation 83.45

Chapter 84: EPSR Commands

Command List 84.2

Part 7: Network Management

Chapter 85: AMF Introduction and Configuration

Introduction to AMF 85.2
AMF Supported Products and Software Versions 85.2
Key Benefits of AMF 85.3
Unified Command-Line 85.3
Configuration Backup and Recovery 85.3
Rolling-Reboot Upgrade 85.3
Node Provisioning 85.4
AMF Terminology and Introduction 85.5
AMF Network 85.5
AMF Nodes 85.5
Node Licensing 85.5
Node Interconnection 85.6
AMF Domains 85.6
AMF Network Operational Concepts 85.8
Retention and Use of the 'Manager' Username 85.8
Working-Set 85.8
AMF Restricted Login 85.9
Loop-Free Data Plane 85.9
Aggregators 85.9
VCStacks 85.9
AMF External Removable Media 85.9
AMF Interaction with QoS and ACLs 85.10
NTP and AMF 85.10
Configuring AMF 85.12
AMF Tunneling (Virtual Links) 85.18
Verifying the AMF Network 85.22
Configuring Multiple Nodes at the Same Time: the Unified CLI 85.24
Working-Set Groups 85.25
Executing Commands on Working-Sets 85.27
Interactive Commands 85.29
AMF Backups 85.30
Using External Media Storage 85.30
Performing a Manual Backup 85.31
Backing up to Remote Servers 85.32
Node Recovery 85.34
Automatic Node Recovery 85.34
Restoring a Node to a "Clean" State 85.36
Manual Node Recovery 85.37
Node Recovery on VCStacks 85.38
AMF Safe Configuration 85.39
Detecting AMF Safe Configuration Operation 85.39
AMF Safe Configuration Procedures 85.39
Undoing an AMF Safe Configuration 85.40
Rolling-Reboot Firmware Upgrade 85.42
Performing a Rolling-Reboot Upgrade 85.44

Chapter 86: AMF Commands

Introduction 86.3
AMF Naming Convention 86.3

Chapter 87: NTP Introduction and Configuration

Introduction 87.2
Overview 87.2
NTP on the Switch 87.3
Troubleshooting 87.4
Configuration Example 87.4

Chapter 88: NTP Commands

Command List 88.2

Chapter 89: Dynamic Host Configuration Protocol (DHCP) Introduction

Introduction 89.2
BOOTP 89.2
DHCP 89.2
DHCP Relay Agents 89.3
Configuring the DHCP Server 89.4
Create the Pool 89.4
Define the Network 89.4
Define the Range 89.5
Set the Lease 89.5
Enable DHCP Leasequery 89.5
Set the Options 89.7
DHCP Lease Probing 89.8
DHCP Relay Agent Introduction 89.9
Configuring the DHCP Relay Agent 89.9
DHCP Relay Agent Information Option (Option 82) 89.10
DHCPv6 Relay Agent Notification for DHCPv6 PD 89.13
Configuring the DHCP Client 89.14
Clearing Dynamically Allocated Lease Bindings 89.14

Chapter 90: Dynamic Host Configuration Protocol (DHCP) Commands

Command List 90.2

Chapter 91: DHCP for IPv6 (DHCPv6) Introduction and Configuration

DHCPv6 Introduction 91.2
DHCPv6 for IPv6 ..... 91.3
DHCPv6 Prefix Delegation 91.3
DHCPv6 RFCs 91.4
DHCPv6 Messages 91.5
DHCPv6 Renewal and Rebinding 91.8
Stateful DHCPv6 Message Exchange 91.9
Stateless DHCPv6 Message Exchange 91.10
DHCPv6 Relay Agent Stateful Message Exchange 91.11
DHCPv6 Prefix Delegation Message Exchange 91.12
DHCPv6 Client and Server Identification 91.13
DHCPv6 Server and Client Functionality 91.14
DHCPv6 Server Functionality 91.14
DHCPv6 Client Functionality 91.15
Configuring DHCPv6 Prefix Delegation 91.17

Configuring the DHCPv6 Server Delegation Pool 91.17
Configuring the DHCPv6 PD Client 91.18
Configuring DHCPv6 Server/Stateful Client (Prefix) 91.19
Configuring DHCPv6 Server/Stateful Client (Range) 91.20
Configuring DHCPv6 Server/Stateless Client 91.21
Configuring DHCPv6 Relay / Server / Client 91.22
Configuring PD Server / PD Client / Stateless Client 91.27
Configuring PD via DHCPv6 Relay 91.29
Configuring PD Subdelegation with SLAAC 91.32
Configuring PD Subdelegation for Multiple VLANs 91.35
Configuring DHCPv6 Relay with Recursive PD Subdelegation 91.38
PD Subdelegation System Configuration 91.42
Stateful_Client Configuration 91.44
Stateless_Client Configuration 91.44
PD_Client Configuration 91.44
DHCPv6_Relay Configuration 91.45
PD_Subdelegation Configuration 91.46
PD_Server1 Configuration 91.47
PD_Server2 Configuration 91.47

Chapter 92: DHCP for IPv6 (DHCPv6) Commands

Command List 92.2

Chapter 93: SNMP Introduction

Introduction 93.2
Network Management Framework 93.2
Structure of Management Information 93.4
Names 93.5
Instances 93.6
Syntax 93.7
Access 93.7
Status 93.7
Description 93.7
The SNMP Protocol 93.8
SNMP Versions 93.8
SNMP Messages 93.9
Polling versus Event Notification 93.9
Message Format for SNMPv1 and SNMPv2c 93.10
SNMP Communities (Version v1 and v2c) 93.11
SNMPv3 Entities 93.11
SNMPv3 Message Protocol Format 93.12
SNMPv1 and SNMPv2c 93.13
SNMP MIB Views for SNMPv1 and SNMPv2c 93.13
SNMP Communities 93.13
Configuration Example (SNMPv1 and v2) 93.15
SNMPv3 ..... 93.18
SNMP MIB Views for SNMPv3 ..... 93.18
SNMP Groups 93.18
SNMP Users 93.18
Configuration Example (SNMPv3) 93.19
Using SNMP to Manage Files and Software 93.20
Copy a File to or from a TFTP Server 93.20
Upgrade Software and Configuration Files 93.22

Chapter 94: SNMP Commands

Command List 94.2

Chapter 95: SNMP MIBs

Introduction 95.2
About MIBs 95.2
About SNMP 95.2
Obtaining MIBs 95.2
Loading MIBs 95.3
Allied Telesis Enterprise MIB 95.5
AT-ALMMON-MIB 95.6
AT-ATMF-MIB 95.8
AT-BOARDS-MIB 95.13
AT-DHCPSN-MIB 95.17
AT-DNS-CLIENT-MIB 95.20
AT-ENVMONv2-MIB 95.21
AT-EPSRv2-MIB 95.29
AT-FILEv2-MIB 95.31
AT-IP-MIB 95.38
AT-LICENSE-MIB 95.40
AT-LOG-MIB 95.43
AT-LOOPPROTECT-MIB 95.45
AT-MIBVERSION-MIB 95.47
AT-NTP-MIB 95.48
AT-PRODUCTS-MIB 95.51
AT-RESOURCE-MIB 95.54
AT-SETUP-MIB 95.55
AT-SMI-MIB 95.63
AT-SYSINFO-MIB 95.65
AT-TRIGGER-MIB 95.69
AT-USER-MIB 95.71
AT-VCSTACK-MIB 95.73
AT-VLANINFO-MIB 95.78
Other Enterprise MIBs 95.80
sFlow-MIB 95.80
Public MIBs 95.81

Chapter 96: LLDP, LLDP-MED and Voice VLAN Introduction and Configuration

Introduction 96.2
Link Layer Discovery Protocol 96.2
LLDP-MED 96.3
Voice VLAN 96.3
LLDP Advertisements 96.4
Type-Length-Value (TLV) 96.4
LLDP-MED: Location Identification TLV 96.7
Transmission and Reception 96.8
LLDP-MED Operation 96.9
Storing LLDP Information 96.10
Configuring LLDP 96.11
Configure LLDP 96.12
Configure LLDP-MED 96.14
Configure Authentication for Voice VLAN 96.19

Chapter 97: LLDP Commands

Introduction 97.2
Command List 97.2

Chapter 98: SMTP Commands

Command List 98.2

Chapter 99: RMON Introduction and Configuration

Introduction 99.2
Overview 99.2
RMON Configuration Example 99.3

Chapter 100: RMON Commands

Command List 100.2

Chapter 101: Triggers Introduction

Introduction 101.2
Trigger Facility 101.2
Configuring a Trigger 101.2
Troubleshooting Triggers 101.5

Chapter 102: Triggers Configuration

Introduction 102.2
Restrict Internet Access 102.2
Capture Unusual CPU and RAM Activity 102.4
See Daily Statistics 102.6
Turn Off Power to Port LEDs 102.7
Reduce Power Supplied to Ports 102.9
Capture Show Output and Save to a USB Storage Device 102.11
Load a Release File From a USB Storage Device 102.12

Chapter 103: Trigger Commands

Command List 103.2

Chapter 104: Ping Polling Introduction and Configuration

Introduction 104.2
How Ping Polling Works 104.2
Configuring Ping Polling 104.4
  Creating a Polling Instance 104.4
  Customizing a Polling Instance 104.5
  Troubleshooting Ping Polling 104.6
Interaction with Other Protocols 104.7

Chapter 105: Ping-Polling Commands

Command List 105.2

Chapter 106: sFlow Introduction and Configuration

sFlow Introduction 106.2
The sFlow Agent 106.3

Sampling Methods 106.3
The sFlow Collector 106.5
Configuring sFlow on your Switch 106.5
Configuration Procedure 106.6
Configuration Example 106.7
sFlow Datagrams 106.12
The sFlow MIB 106.13

Chapter 107: sFlow Commands

Command List 107.2

Chapter 108: Cable Fault Locator Introduction

Introduction to the Cable Fault Locator 108.2
Capabilities 108.2
TDR Operating Principles 108.2
Using the Cable Fault Locator 108.3

Chapter 109: Cable Fault Locator Commands

Command List 109.2

Part 8: Virtual Chassis Stacking

Chapter 110: VCStack Introduction

VCStack Introduction 110.2
Features of Virtual Chassis Stacking 110.2
VCStack Capable Switches 110.2
The Physical Stack 110.3
Two-Switch Stack Configuration 110.3
Multiple Switch Stack Configuration 110.4
Resilient Stacked Topology 110.4
Stack Formation 110.8
The Role of the Stack Master 110.8
Stack Management VLAN 110.9
Stack Member Failure and Recovery 110.12
Fixed or Virtual MAC Addressing 110.12
Stack Resiliency Link 110.13
Stack Failure Recovery 110.14
Stack Separation and Recovery 110.15
Stack Maintenance 110.15
Disabled Master Monitoring (DMM) 110.17
Provisioning (Stack Members) 110.19
Provisioned Board Classes 110.19
Applying Hardware Provisioning 110.19
Removing Hardware Provisioning 110.21
Displaying Provisioned Configurations 110.22
Provisioning and Configuration Management 110.23
Software Version Auto Synchronization 110.24
Introduction 110.24
How Auto Synchronization Works 110.24
Stack License Management 110.27

Chapter 111: Stacking Commands

Introduction....111.2
Command List....111.3

Appendix A: Command List

Appendix B: Changes in this Software Version

Appendix C: GUI Reference

Appendix D: Glossary

Part 1: Setting up the Switch

- Chapter 1 Getting Started
- Chapter 2 Command Syntax Conventions in this Software Reference
- Chapter 3 Start-Up Sequence
- Chapter 4 CLI Navigation Commands
- Chapter 5 User Access Commands
- Chapter 6 Creating and Managing Files
- Chapter 8 Licensing Introduction and Configuration
- Chapter 9 Licensing Commands
- Chapter 10 System Configuration and Monitoring Commands
- Chapter 11 Debugging and Logging
- Chapter 12 Logging Commands
- Chapter 13 Scripting Commands
- Chapter 14 Interface Commands
- Chapter 15 Interface Testing Commands

Chapter 1: Getting Started

Introduction 1.2
How to Login 1.2
How to Get Command Help 1.3
Viewing a List of Valid Parameters 1.3
Completing Keywords 1.5
Viewing Command Error Messages 1.6
How to Work with Command Modes 1.7
Entering Privileged Exec Commands when in a Configuration Mode 1.9
How to See the Current Configuration 1.10
Default Settings 1.11
The Default Configuration 1.12
How to Change the Password 1.13
How to Set Strong Passwords 1.14
How to Set an IP Address on VLAN 1 ..... 1.16
How to Save and Boot from the Current Configuration 1.17
How to Save to the Default Configuration File 1.17
How to Create and Use a New Configuration File 1.17
How to Return to the Factory Defaults 1.19
How to See System Information 1.20
Viewing Overall System Information 1.20
Viewing Voltage, Fan Status, and Temperature 1.20
Viewing the Serial Number 1.21
How to Set System Parameters 1.21
How to Change the Telnet Session Timeout 1.21
How to Name the Switch 1.23
How to Display a Text Banner at Login 1.24
How to Set the Time and Date 1.25
How to Show Current Settings 1.25
How to Set the Time and Date 1.25
How to Set the Timezone 1.26
How to Configure Summer-Time 1.26
How to Add and Remove Users 1.27
Pre-Encrypted Passwords 1.28
How to Undo Settings 1.30
How to Use the *no* Parameter 1.30
How to Use the *default* Parameter 1.30
How to Upgrade the Firmware 1.31
Save Power with the Eco-Friendly Feature 1.32
The Eco-Friendly LED Feature 1.32
The Eco-Friendly LPI Feature 1.32
Trouble-shoot Fiber and Pluggable Issues 1.33
Using the Find Me feature 1.34
Continuous Reboot Prevention 1.35
Controlling “show” Command Output 1.37
AlliedWare Plus GUI 1.39

Introduction

This chapter introduces a number of commonly-used management features of the AlliedWare Plus™ Operating System (OS).

How to Login

Step 1: Set the console baud rate if necessary

The default baud rate is 9600.

By default the AlliedWare Plus™ OS supports VT100 compatible terminals on the console port. This means that the terminal size is 80 columns by 24 rows.

Step 2: Login with manager/friend

The defaults are:

```
username: manager
password: friend
```

The switch logs you into User Exec mode. From User Exec mode, you can perform high-level diagnostics (some **show** commands, ping, traceroute, for example), start sessions (Telnet, SSH), and change mode.

How to Get Command Help

The following kinds of command help are available:

- lists of valid parameters with brief descriptions (the ? key)
- completion of keywords (the Tab key)
- error messages for incomplete or incorrect syntax

Command Abbreviations

The AlliedWare Plus™ CLI contains a number of abbreviations for its commands. For example, the **show interface** command can be entered in the abbreviated form shown below:

```
awplus# sh in vlan100
```

Viewing a List of Valid Parameters

To get syntax help, type ? (i.e. “space question mark”) after:

- the prompt. This will list all commands available in the mode you are in.
- one or more parameters. This will list parameters that can come next in the partial command.
- one or more letters of a parameter. This will list matching parameters.

Note

The AlliedWare Plus™ OS only displays one screenful of text at a time, with the prompt “--More--” at the end of each screenful. Press the space bar to display the next screenful or the Q key to return to the command prompt.

Example

To see which commands are available in Privileged Exec mode, enter “?” at the Privileged Exec mode command prompt:

```
awplus# ?
```

This results in output as shown in **Figure 1-1** below:

Figure 1-1: Example output from the ? command

```
Exec commands:
activate          Activate a script
cd                Change the current working directory
clear             Reset functions
clock             Manage clock
configure         Enter configuration mode
copy              Copy from one file to another
.
.
.
```

Example

To see which **show** commands that start with “i” are available in Privileged Exec mode, enter “?” after **show i**:

```
awplus# show i?
```

This results in the following output:

Figure 1-2: Example output from the show i? command

```
interface          Select an interface to configure
ip                 Internet Protocol (IP)
ipv6               Internet Protocol version 6 (IPv6)
```

Examples To use the ? help to work out the syntax for the **clock timezone** command, enter the following sequence of commands:

```
awplus(config)# clock ?
```

```
summer-time  Manage summer-time
timezone     Set clock timezone
```

```
awplus(config)# clock timezone ?
```

```
TIMEZONE  Timezone name, up to 5 characters
```

```
awplus(config)# clock timezone NZST ?
```

```
minus  negative offset
plus   positive offset
```

```
awplus(config)# clock timezone NZST plus ?
```

```
<0-12>  Time zone offset to UTC
```

```
awplus(config)# clock timezone NZST plus 12
```

The above example demonstrates that the ? help only indicates what you can type **next**. For commands that have a series of parameters, like **clock timezone**, the ? help does not make the number of parameters obvious.

Completing Keywords

To complete keywords, type the Tab key after part of the command.

If only one keyword matches the partial command, the AlliedWare Plus™ OS fills in that keyword. If multiple keywords match, it lists them.

Examples In this example we use Tab completion in successive steps to build the complete command **show ip dhcp snooping**. We have included "<Tab>" to show where to type the Tab key — this is not displayed on screen.

Figure 1-3: Example output after entering the command show ip <Tab>

```
as-path-access-list  bgp                  community-list
dhcp                 dhcp-relay           domain-list
domain-name          extcommunity-list    filter
forwarding           igmp                 interface
irdp                 mroute               mvif
name-server          nat                  ospf
pim                  protocols            rip
route                rpf
```

Figure 1-4: Example output after entering the command, show ip d<Tab>

```
awplus#show ip d<Tab>
dhcp         dhcp-relay       domain-list      domain-name
```

Figure 1-5: Example output from the show ip dhcp <Tab> command

```
awplus#show ip dhcp <Tab>
binding   pool      server    snooping
```

Figure 1-6: Example output from the show ip dhcp s<Tab> command

```
awplus#show ip dhcp s<Tab>
server    snooping
```

Viewing Command Error Messages

The switch displays the following generic error messages about command input:

% Incomplete command—this message indicates that the command requires more parameters. Use the ? help to find out what other parameters are available.

```
awplus# interface
```

```
% Incomplete command.
```

% Invalid input detected at '^' marker—this indicates that the switch could not process the command you entered. The switch also prints the command and marks the first invalid character by putting a '^' under it. Note that you may get this error if you enter a command in the wrong mode, as the following output shows.

```
awplus# interface port1.0.1
```

```
interface port1.0.1
          ^
% Invalid input detected at '^' marker.
```

% Unrecognized command—when you try to use ? help and get this message, it indicates that the switch can not provide help on the command because it does not recognize it. This means the command does not exist, or that you have entered it in the wrong mode, as the following output shows.

```
awplus# interface ?
```

```
% Unrecognized command
```

Note

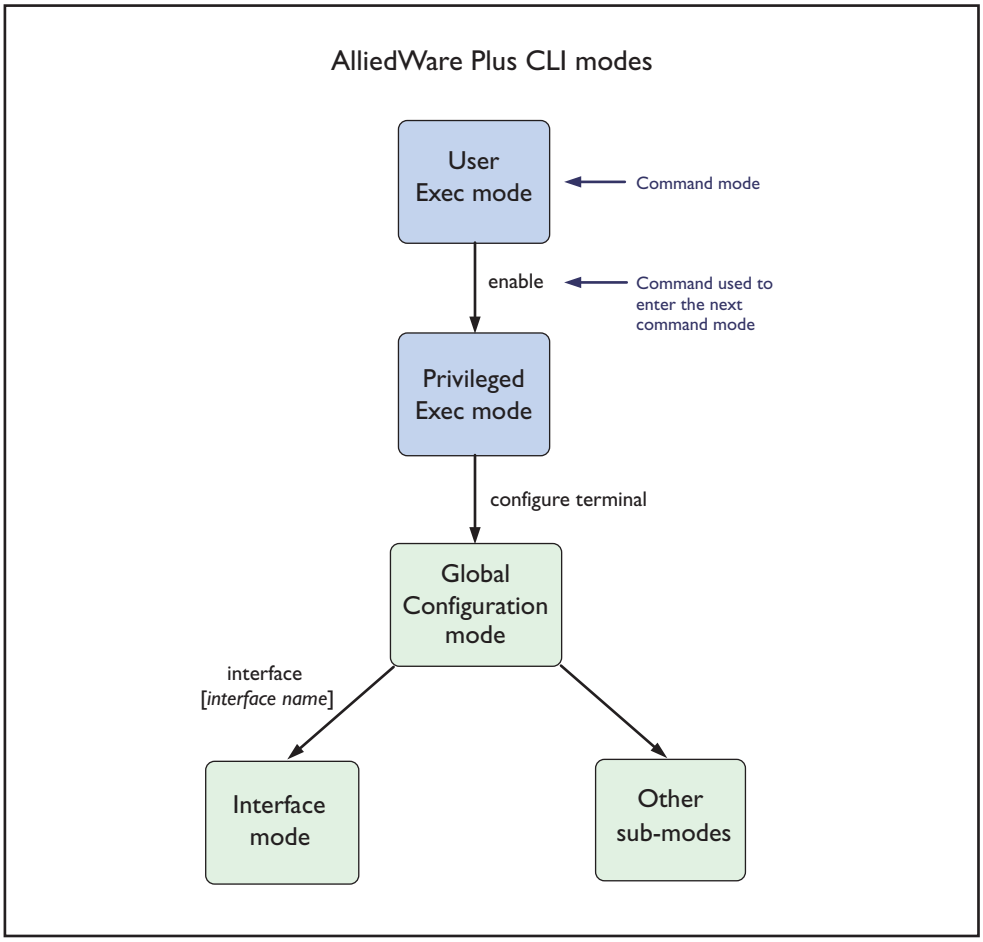
The AlliedWare Plus™ OS does not tell you when commands are successful. If it does not display an error message, you can assume the command was successful.

How to Work with Command Modes

The following figure shows the command mode hierarchy and the commands you use to move to lower-level modes.

Multiple users can telnet and issue commands using the User Exec mode and the Privileged Exec mode. However, only one user is allowed to use the Configure mode at a time. This prevents multiple users from issuing configuration commands simultaneously.

Figure 1-7: AlliedWare Plus™ CLI modes



User Exec mode

User Exec mode is the mode you log into on the switch.

It lets you perform high-level diagnostics (**show** commands, ping, traceroute, for example), start sessions (Telnet, SSH), and change mode.

The default User Exec mode prompt is **awplus>**

Privileged Exec mode

To change from User Exec to Privileged Exec mode, enter the command:

```
awplus> enable
```

Privileged Exec mode is the main mode for monitoring—for example, running **show** commands and debugging. From Privileged Exec mode, you can do all the commands from User Exec mode plus many system commands.

The default Privileged Exec mode prompt is **awplus#**

Global Configuration mode

To change from Privileged Exec to Global Configuration mode, enter the command:

```
awplus# configure terminal
```

From Global Configuration mode, you can configure most aspects of the switch.

The default Global Configuration mode prompt is **awplus(config)#**

Lower-level configuration modes

A number of features are configured by entering a lower-level mode from Global Configuration mode.

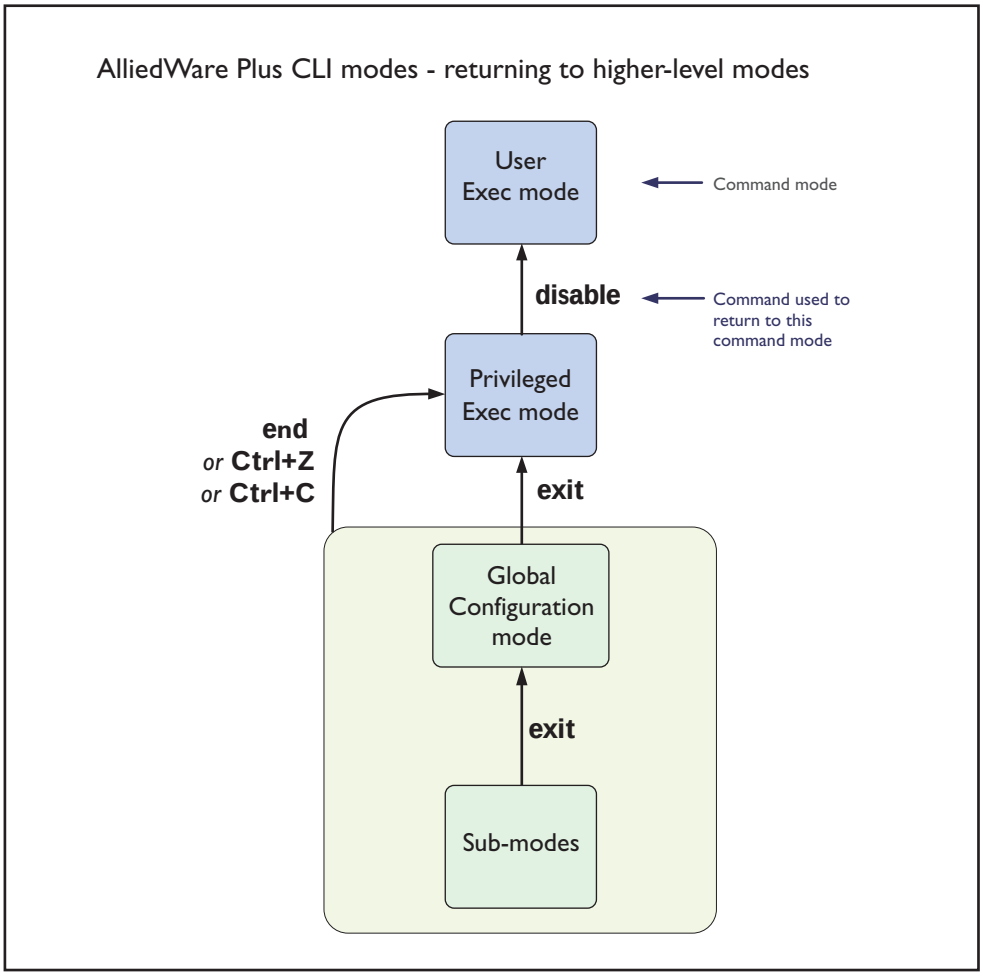
Some protocols have commands in both Global Configuration mode and lower-level configuration modes. For example, to configure MSTP, you use:

- Global Configuration mode to select MSTP as the spanning tree mode
- MST mode to create instances and specify other MSTP settings
- Interface Configuration mode to associate the instances with the appropriate ports.

Returning to higher-level modes

The following figure shows the commands to use to move from a lower-level mode to a higher-level mode.

Figure 1-8: Returning to higher-level modes



Examples To go from Interface Configuration to Global Configuration mode:

```
awplus(config-if)# exit
   awplus(config)#
```

To go from Interface Configuration to Privileged Exec mode:

```
awplus(config-if)# end
            awplus#
```

To go from Privileged Exec to User Exec:

```
            awplus# exit
            awplus>
```

Entering Privileged Exec Commands when in a Configuration Mode

As you configure the switch you will be constantly entering various **show** commands to confirm your configuration. This requires constantly changing between configuration modes and Privileged Exec mode.

However, you can run Privileged Exec commands without changing mode, by using the command:

```
do <command you want to run>
```

You cannot use the ? help to find out command syntax when using the **do** command.

Example To display information about the IP interfaces when in Global Configuration mode, enter the command:

This results in the following output:

```
awplus(config)# do show ip int brief
```

Figure 1-9: Example output after entering the command "do show ip int brief"

```
Interface          IP-Address      Status          Protocol
vlan1              unassigned      admin up        running
vlan2              unassigned      admin up        running
```

How to See the Current Configuration

The current configuration is called the running-config. To see it, enter the following command in either Privileged Exec mode or any configuration mode:

```
awplus# show running-config
```

To see only part of the current configuration, enter the command:

```
awplus# show running-config|include <word>
```

This displays only the lines that contain *word*.

To start the display at a particular place, enter the command:

```
awplus# show running-config|begin <word>
```

This searches the running-config for the first instance of *word* and begins the display from that line.

Note The **show running-config** command works in all modes except User Exec mode.

Default Settings

When the switch first starts up with the AlliedWare Plus™ OS, it applies default settings and copies these defaults dynamically into its running-config.

These default settings mean that the AlliedWare Plus™ OS:

- encrypts passwords, such as user passwords
- records log message priority in log messages
- turns on the telnet server so that you can telnet to the switch
- enables the switch to look up domain names (but for domain name lookups to work, you have to configure a DNS server)
- turns off L3 multicast packet switching in the switch's hardware. This prevents L3 multicast from flooding the switch's CPU in its default state as an L2 switch
- sets the maximum number of ECMP routes to 4
- turns on RSTP on all ports. Note that the ports are not set to be edge ports
- sets all the switch ports to access mode. This means they are untagged ports, suitable for connecting to hosts
- creates VLAN 1and adds all the switch ports to it
- allows logins on the serial console port
- allows logins on VTY sessions (for telnet, for example)
- has switching enabled, so Layer 2 traffic is forwarded appropriately without further configuration
- allocates all the routing table memory space to IPv4 and IPv6 routes
- has ports set to autonegotiate their speed and duplex mode
- has copper ports set to auto MDI/MDI-X mode

The Default Configuration

Most of the above default settings are in the form of commands, which the switch copies to its running-config when it first boots up.

For more information about start-up files, see **"How to Save and Boot from the Current Configuration" on page 1.17**.

How to Change the Password

To change the password for the manager account, enter Global Configuration mode and enter the following command:

```
awplus(config)# username manager password <new-password>
```

The password can be up to 23 characters in length and include characters from up to four categories. The password categories are:

- uppercase letters: A to Z
- lowercase letters: a to z
- digits: 0 to 9
- special symbols: all printable ASCII characters not included in the previous three categories. The question mark ? cannot be used as it is reserved for help functionality.

How to Set Strong Passwords

The password security rules are disabled by default. To set password security rules for users with administrative rights, or privilege level 15, enter Global Configuration mode.

You can then either specify whether the user is forced to change an expired password at the next login, or specify whether the user is not allowed to login with an expired password. You will need to specify a password lifetime greater than 0 before selecting either of these features. Note that the **security-password forced-change** and the **security-password reject-expired-pwd** commands cannot be enabled concurrently.

Password lifetime

Enter the following command to specify the password lifetime in days:

```
awplus(config)# security-password lifetime <0-1000>
```

Note that the value 0 will disable lifetime functionality and passwords will never expire. If lifetime functionality is disabled, the **security-password forced-change** command and the **security-password warning** command are also disabled.

Password forced change

To specify that a user is forced to change an expired password at the next login, enter the following command:

```
awplus(config)# security-password forced-change
```

If the **security-password forced-change** command is enabled, users with expired passwords are forced to change to a password that must comply with the current password security rules at the next login.

Reject expired password

To specify that a user is not allowed to login with an expired password, enter the following command:

```
awplus(config)# security-password reject-expired-pwd
```

If the **security-password reject-expired-pwd** command is enabled, users with expired passwords are rejected at login. Users then have to contact the Network Administrator to change their password.

Caution

Once all users' passwords are expired you are unable to login to the device again if the security-password reject-expired-pwd command has been executed. You will have to reboot the device with a default configuration file, or load an earlier software version that does not have the security password feature.
We recommend you never have the command line "security-password reject-expired-pwd" in a default config file.

Use other password security rules to further configure password security settings.

Password warning

To specify the number of days before the password expires that the user will receive a warning message specifying the remaining lifetime of the password, enter the command:

```
awplus(config)# security-password warning <0-1000>
```

The value 0 will disable warning functionality and the warning period must be less than, or equal to, the password lifetime.

Password history

To specify the number of previous passwords that are unable to be reused enter the command:

```
awplus(config)# security-password history <0-15>
```

The value `0` will disable history functionality. If history functionality is disabled, all users' password history is reset and all password history is lost. A new password is invalid if it matches a password retained in the password history.

Password minimum length

To specify the minimum allowable password length, enter the command:

```
awplus(config)# security-password minimum-length <1-23>
```

Password minimum categories

To specify the minimum number of categories that the password must contain in order to be considered valid, enter the command:

```
awplus(config)# security-password minimum-categories <1-4>
```

The password categories are:

- uppercase letters: A to Z
- lowercase letters: a to z
- digits: 0 to 9
- special symbols: all printable ASCII characters not included in the previous three categories. The question mark ? cannot be used as it is reserved for help functionality

To ensure password security, the minimum number of categories should align with the lifetime selected, i.e. the fewer categories specified the shorter the lifetime specified.

How to add a user is described in **"How to Add and Remove Users" on page 1.27**.

Display security password settings

To list the configuration settings for the various security password rules, enter the command:

```
awplus(config)# show security-password configuration
```

To list users remaining lifetime or last password change, enter the command:

```
awplus(config)# show security-password user
```

How to Set an IP Address on VLAN 1

This section describes how to set an IP address on the default VLAN (**vlan1**).

Step 1: If desired, check the current configuration

After logging in, enter Privileged Exec mode by using the command:

```
awplus># enable
```

Then check the current configuration by using one of the following commands:

```
awplus# show ip interface vlan1 brief
```

The output looks like this:

```
Interface            IP-Address      Status          Protocol
vlan1                172.28.8.200    admin up        running
```

```
awplus# show running-config interface vlan1
```

This output looks like this:

```
!
interface vlan1
 ip address 172.28.8.200/16
!
```

Step 2: Enter Interface Configuration mode for the vlan1 interface

Enter Global Configuration mode and enter the command:

```
awplus(config)# interface vlan1
```

Step 3: Enter the IP address and mask

Enter the command:

```
awplus(config-if)# ip address <address/mask>
```

For example, to set the address to 172.28.8.210/16, enter the command:

```
awplus(config-if)# ip address 172.28.8.210/16
```

How to Save and Boot from the Current Configuration

This section tells you how to save your configuration and run the saved configuration when the switch starts up.

You can either:

- save the configuration to the switch's default configuration file (called "default.cfg"). By default, the switch uses that file at start-up.
- create a new configuration file and set the switch to use the new configuration file at start-up.

How to Save to the Default Configuration File

Enter Privileged Exec mode and enter the command:

```
awplus# copy running-config startup-config
```

The parameter **startup-config** is a short-cut for the current boot configuration file, which will be the default configuration file unless you have changed it, as described in the next section.

How to Create and Use a New Configuration File

Step 1: Copy the current configuration to a new file

Enter Privileged Exec mode and enter the command:

```
awplus# copy running-config <destination-url>
```

Example To save the current configuration in a file called `example.cfg`, enter the command

```
awplus# copy running-config example.cfg
```

Step 2: Set the switch to use the new file at startup

To run the new file's configuration when the switch starts up, enter Global Configuration mode and enter the command:

```
awplus(config)# boot config-file <filepath-filename>
```

Note that you can set the switch to use a configuration file on a USB storage device if you have saved the configuration file to a USB storage device. You can only specify that the configuration file is on a USB storage device if there is a backup configuration file already specified in Flash.

To set a backup configuration file to load if the main configuration file cannot be loaded, enter the command:

```
awplus(config)# boot config-file backup <filepath-filename>
```

For an explanation of the configuration fallback order, see **"The Configuration File Fallback Order" on page 6.10**.

Example To run the commands in `example.cfg` on startup, enter the command:

```
awplus(config)# boot config-file flash:/example.cfg
```

To set `backup.cfg` as the backup to the main configuration file, enter the command:

```
awplus(config)# boot config-file backup flash:/backup.cfg
```

Step 3: Display the new settings

To see the files that the switch uses at startup, enter Privileged Exec mode and enter the command:

```
awplus# show boot
```

The output looks like this:

```
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : flash:/x510-5.4.4-3.5.rel
Backup  boot image : flash:/x510-5.4.4-3.5.rel
Default boot config: flash:/default.cfg
Current boot config: usb:/example.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file exists)
```

Step 4: Continue updating the file when you change the configuration

When you next want to save the current configuration, enter Privileged Exec mode and enter the command:

```
awplus# copy running-config startup-config
```

The parameter **startup-config** is a short-cut for the current boot configuration file.

How to Return to the Factory Defaults

The switch dynamically adds the default settings to the running-config at start-up if the default file is not present. This section describes how to use this feature to return to the factory defaults.

Note

After reboot the show running-config output will show the default factory settings for your switch once you have removed the default.cfg file. To recreate the default.cfg file enter copy running-config startup-config. When you enter copy running-config startup-config commands the default.cfg file is updated with the startup-config.

Completely restore defaults

To completely remove your configuration and return to the factory default configuration, delete or rename the default file and make sure no other file is set as the start-up configuration file.

To find the location of the default boot configuration file, enter Privileged Exec mode and enter the command:

```
awplus# show boot
```

To delete the default file when it is the current boot configuration file, enter Privileged Exec mode and enter either of the commands:

```
awplus# delete force <filename>
```

or:

```
awplus# erase startup-config
```

Note that erasing startup-config deletes the current boot configuration file—it does not simply stop the file from being the boot file.

To make sure that no other file is loaded at start-up, enter Global Configuration mode and enter the command:

```
awplus(config)# no boot config-file
```

Partially restore defaults

To partially restore the default settings, make a configuration file that contains the settings you want to keep and set this as the start-up configuration file. On start-up, the switch will add the missing settings to the running-config.

How to See System Information

This section describes how to view the following system information:

- overview information
- details of temperature and voltage
- serial number

Viewing Overall System Information

To display an overview of the switch hardware, software, and system settings, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show system
```

The output looks like this:

```
Switch System Status                                    Tue Aug 14 14:30:08 2012
Board       ID  Bay   Board Name                              Rev  Serial number
-------------------------------------------------------------------------------
Base       369        x510-28GTX                             X2-1  A04736H120500033
-------------------------------------------------------------------------------
RAM:  Total: 485460 kB Free: 394104 kB
Flash: 63.0MB Used: 20.1MB Available: 42.9MB
-------------------------------------------------------------------------------
Environment Status : Normal
Uptime             : 17 days 19:00:46
Bootloader version : 2.0.9-devel


Current software   : x510-5.4.4-3.5.rel
Software version   : Version 5.4.4-2
Build date         : Tue Apr 24 13:42:56 NZST 2012

Current boot config: flash:/default.cfg (file exists)

System Name
 awplus
```

Viewing Voltage, Fan Status, and Temperature

The switch monitors the environmental status of the switch and its power supplies and fan. To display this information, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show system environment
```

The output looks like this:

```
awplus#show system environment
Stack Environment Monitoring Status

Stack member 1:

Overall Status: Normal

Resource ID: 1  Name: x510-28GTX
ID  Sensor (Units)                    Reading  Low Limit High Limit Status
1   Fan: Fan 1 (Rpm)                     4344       3000          -      Ok
2   Voltage: 1.8V (Volts)               1.804      1.617      1.978     Ok
3   Voltage: 1.0V (Volts)               0.995      0.896      1.099     Ok
4   Voltage: 3.3V (Volts)               3.291      2.960      3.613     Ok
5   Voltage: 5.0V (Volts)               5.066      4.477      5.498     Ok
6   Voltage: 1.2V (Volts)               1.187      1.072      1.318     Ok
7   Temp: CPU (Degrees C)                  50        -10         90     Ok
```

Viewing the Serial Number

The switch's serial number is displayed in the output of the **show system** command on page 10.47, but for convenience, you can also display it by itself. To do this, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show system serialnumber
```

The output looks like this:

```
P1FY7502C
```

How to Set System Parameters

You can set system parameters to personalize the switch and make it easy to identify it when troubleshooting. This section describes how to configure the following system parameters:

- telnet session timeout
- switch name
- login banner

How to Change the Telnet Session Timeout

By default, telnet sessions time out after 10 minutes of idle time. If desired, you can change this.

To change the timeout for all telnet sessions, enter Global Configuration mode and enter the commands:

```
awplus(config)# line vty 0 32
awplus(config-line)# exec-timeout <new-timeout>
```

The new timeout value only applies to new sessions, not current sessions.

Examples To set the timeout to 30 minutes, enter the command:

```
awplus(config-line)# exec-timeout 30
```

To set the timeout to 30 seconds, enter the command:

```
awplus(config-line)# exec-timeout 0 30
```

To set the timeout to infinity, so that sessions never time out, enter either of the commands:

```
awplus(config-line)# no exec-timeout
awplus(config-line)# exec-timeout 0 0
```

How to Name the Switch

To give the switch a name, enter Global Configuration mode and enter the command:

```
awplus(config)# hostname <name>
```

For example, to name the switch “mycompany”:

```
awplus(config)# hostname mycompany
```

The prompt displays the new name:

```
mycompany(config)#
```

The name can contain hyphens and underscore characters, for example:

```
mycompany(config)#hostname mycompany_more_words
mycompany_more_words(config)#hostname mycompany-hyphenated
mycompany-hyphenated(config)
```

However, the name must be a single word, as the following example shows.

```
mycompany(config)#hostname mycompany more words
                                   ^
% Invalid input detected at '^' marker.
```

It also cannot be surrounded by quote marks, as the following example shows.

```
awplus(config)#hostname "mycompany more words"
% hostname contains invalid characters
```

Removing the name

To remove the hostname, enter the command:

```
mycompany(config)# no hostname
```

The prompt changes back to the default prompt:

```
awplus(config)#
```

How to Display a Text Banner at Login

By default, the switch displays the AlliedWare Plus™ OS version and build date before login. You can customize this by changing the Message of the Day (MOTD) banner.

To enter a new MOTD banner, enter Global Configuration mode and enter the command:

```
awplus(config)# banner motd <banner-text>
```

The text can contain spaces and other printable characters. You do not have to surround words with quote marks.

Example

To display "this is a new banner" when someone logs in, enter the command:

```
awplus(config)# banner motd this is a new banner
```

This results in the following output at login:

```
awplus login: manager
Password:
this is a new banner
awplus>
```

Removing the banner

To return to the default banner (AlliedWare Plus™ OS version and build date), enter the command:

```
awplus(config)# banner motd default
```

To remove the banner instead of replacing it, enter the command:

```
awplus(config)# no banner motd
```

How to Set the Time and Date

There are three aspects to setting the time and date:

- setting the current time and date (**"How to Set the Time and Date" on page 1.25**)
- setting the timezone (**"How to Set the Timezone" on page 1.26**)
- configuring the switch to automatically change the time when summer-time begins and ends (**"How to Configure Summer-Time" on page 1.26**)

Instead of manually setting the time, you can use NTP to automatically get the time from another device.

How to Show Current Settings

To display the current time, timezone and date, enter Privileged Exec mode and enter the command:

```
awplus# show clock
```

The output looks like this:

```
UTC Time:   Wed,  16 May 2013 16:08:14 +0000
Timezone: UTC
Timezone Offset: +00:00
Summer time zone: None
```

How to Set the Time and Date

To set the time and date, enter Privileged Exec mode and enter the **clock set** command:

```
clock set <hh:mm:ss> <day> <month> <year>
```

:where:

- *hh* is two digits giving the hours in 24-hour format (e.g. **14**)
- *mm* is two digits giving the minutes
- *ss* is two digits giving the seconds
- *day* is two digits giving the day of the month
- *month* is the first three letters of the month name (e.g. **sep**)
- *year* is four digits giving the year

Example To set the time to 14:00:00 on 25 January 2012, use the command:

```
awplus# clock set 14:00:00 25 jan 2012
```

How to Set the Timezone

To set the timezone, enter Global Configuration mode and enter the **clock timezone** command:

```
clock timezone <timezone-name> {plus|minus} <0-12>
```

The <*timezone-name*> can be any string up to 6 characters long.

To return the timezone to UTC+0, enter the command:

```
awplus(config)# no clock timezone
```

Example To set the timezone to Eastern Standard Time, use the command:

```
awplus(config)# clock timezone EST minus 5
```

How to Configure Summer-Time

There are two approaches for setting summer-time:

- *recurring*, when you specify the week when summer-time starts and ends and each year the switch changes the time at those weeks. For example, Eastern Daylight Time (EDT) starts at 2 am on the second Sunday in March and ends at 2 am on the first Sunday in November.
- *date-based*, when you specify the start and end dates for summer-time for a particular year. For example, Eastern Daylight Time (EDT) starts at 2 am on Sunday, 8 March 2008 and ends at 2 am on Sunday, 2 November 2008.

Recurring To set summer-time with recurring dates, enter Global Configuration mode and enter the **clock summer-time recurring** command:

```
clock summer-time <zone-name> recurring <start-week> <start-
day> <start-month> <start-time> <end-week> <end-day>
<end-month> <end-time> <1-180>
```

The <*zone-name*> can be any string up to 6 characters long.

The <*start-time*> and <*end-time*> are in the form `hh:mm`, in 24-hour time.

Note that if you specify 5 for the week, this changes the time on the last day of the month, not the 5th week.

Example To configure EDT, enter the command:

```
awplus(config)# clock summer-time EDT recurring 2 Sun Mar
                02:00 1 Sun Nov 02:00 60
```

Date-based To set summer-time for a single year, enter Global Configuration mode and enter the **clock summer-time date** command:

```
clock summer-time <zone-name> date <start-day> <start-month>
<start-year> <start-time> <end-day> <end-month> <end-year>
<end-time> <1-180>
```

The <*zone-name*> can be any string up to 6 characters long.

The <*start-time*> and <*end-time*> are in the form `hh:mm`, in 24-hour time.

Example For example, to configure EDT for 2008 enter the command:

```
awplus(config)# clock summer-time EDT date 8 Mar 2008 02:00
                2 Nov 2008 02:00 60
```

How to Add and Remove Users

Adding users To add a new user with administrative rights, enter Global Configuration mode and enter the command:

```
awplus(config)# username <name> privilege 15 password
                <password>
```

Both <*name*> and <*password*> can contain any printable character and are case sensitive.

When you add a user with administrative rights, <*password*> will have to conform to the following rules:

- **security-password minimum-categories** command on page 5.20
- **security-password minimum-length** command on page 5.21
- **security-password history** command on page 5.17. If this command is enabled, <*password*> is invalid if it matches a password retained in the password history.

The AlliedWare Plus™ OS gives you a choice of **1** or **15** for the privilege level. Level **1** users are limited to User Exec mode so you need to set most users to level **15**.

For example, to add user Bob with password 123$%^, enter the command:

```
awplus(config)# username Bob privilege 15 password 123$%^
```

Removing users To remove a user, enter Global Configuration mode and enter the command:

```
no username <name>
```

For example, to remove user Bob, enter the command:

```
awplus(config)# no username Bob
```

Note that you can delete all users, including the user called “manager” and the user you are logged in as. If all privilege **15** user accounts are deleted, a warning message is generated:

```
% Warning: No privileged users exist.
```

If all privilege level **15** user accounts are deleted, and there are no other users configured for the device, you may have to reboot with the default configuration file.

If there is a user account on the device with a lower privilege level and a password has already been set with the **enable password** command on page 5.4, you can login and still enter privileged mode. When executing the **enable** command, enter the password created with the **enable password** command. For example, if the password is mypassword:

```
awplus> enable mypassword

awplus#
```

Displaying users

To list the currently logged-in users, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show users
```

The output looks like this:

```
Line    User            Host(s)  Idle      Location        Priv Idletime Timeout
con 0   manager         idle     00:00:00  ttyS0           15   10       N/A
vty 0   bob             idle     00:00:03  172.16.11.3     1    0        5
```

To list all configured users, enter User Exec or Privileged Exec mode and enter the command:

```
awplus# show running-config|include username
```

The output looks like this:

```
username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0
username Bob privilege 15 password 8 $1$gXJLY8dw$iqkMXLgQxbzSOutNUa5E2.
```

Pre-Encrypted Passwords

The running-config output above includes the number 8 after the **password** parameter. This indicates that the password is displayed in its encrypted form.

You can enter the number 8 and a pre-encrypted password on the command line. You may want to pre-encrypt passwords if you need to load them onto switches via an insecure method (such as HTTP, or by emailing them to remote users).

Caution Only enter the number 8 if you are entering a pre-encrypted password—otherwise, you will be unable to log in using the password and will be unable to access the switch through that username. The next section describes why.

Testing this feature

If you want to test the effect of this, *create a new user* for the test instead of using the manager user. The test stops you from logging in as the test user, so you need to have the manager user available to log in as.

The following output shows how specifying the number 8 puts the password into the running-config exactly as you typed it:

```
awplus(config)#username Bob privilege 15 password 8 friend
awplus(config)#show running-config |include username Bob
username Bob privilege 15 password 8 friend
```

After entering the command above, logging in as “Bob” with a password of “friend” does not work. This is because the switch takes the password you enter (“friend”), hashes it, and compares the hash with the string in the running-config (“friend”). The hashed value and “friend” are not the same, so the switch rejects the login.

How to Undo Settings

There are two possibilities for undoing settings: the **no** parameter and the **default** parameter.

How to Use the *no* Parameter

To undo most settings, simply re-enter the first parameters of the configuration command with the parameter **no** before them.

Example You can set the timezone to Eastern Standard Time by entering the command:

```
awplus(config)# clock timezone EST minus 5
```

To remove the timezone setting, enter the command:

```
awplus(config)# no clock timezone
```

How to Use the *default* Parameter

Some commands have a **default** parameter that returns the feature to its default setting.

Example You can change the login banner to “this is a new banner” by entering the command:

```
awplus(config)# banner motd this is a new banner
```

To return to the default banner, enter the command:

```
awplus(config)# banner motd default
```

Note that this command also has a **no** parameter that lets you remove the banner altogether.

How to Upgrade the Firmware

New releases of the AlliedWare Plus™ OS become available regularly. Contact your customer support representative for more information.

Step 1: Put the new release onto your TFTP server or your USB drive

Step 2: If necessary, create space in the switch's Flash memory for the new release

Note that you cannot delete the current release file.

To see how much space is free, use the command:

```
awplus# show file systems
```

Step 3: Copy the new release from your TFTP server or your USB drive onto the switch

Follow the relevant instructions in "Copying with Trivial File Transfer Protocol (TFTP)" on page 6.16, or "Copying to and from NVS or a USB storage device" on page 6.15.

Step 4: Set the switch to boot from the new release

Enter Global Configuration mode and enter the command:

```
awplus(config)# boot system <filepath-filename>
```

You can set a backup release file to load if the main release file cannot be loaded. Enter the command:

```
awplus(config)# boot system backup <filepath-filename>
```

Step 5: Check the boot settings

Enter Privileged Exec mode and enter the command:

```
awplus# show boot
```

Step 6: Reboot

Enter Privileged Exec mode and enter the command:

```
awplus# reboot
```

Save Power with the Eco-Friendly Feature

The Eco-Friendly LED Feature

You can conserve power by enabling the eco-friendly LED (Light Emitting Diode) feature either by entering the **ecofriendly led** command on page 10.13 or by pressing the **eco**-friendly button on the front panel. This feature disables power to the port LEDs, but not the power indicator LED. In the eco-friendly mode, one of the horizontal segments of the seven segment display will glow permanently. If the device is a stack master, the upper segment will glow; if the device is a stack member or a stand alone unit, the center segment will glow.

When the eco-friendly LED feature is enabled, a port's LED will not change if the port's status changes. When the eco-friendly feature is disabled and power is returned to port LEDs, the LEDs will correctly show the current state of the ports.

In a stack environment, enabling the eco-friendly LED feature on the stack master will apply the feature to every member of the stack.

The eco-friendly LED feature is disabled by default. To globally enable the feature for all LED ports on the switch, either push the eco-switch button or enter the commands:

```
awplus# configure terminal
awplus(config)# ecofriendly led
```

For an example of how to configure a trigger to enable the eco-friendly LED feature, see **"Turn Off Power to Port LEDs" on page 102.7**.

The Eco-Friendly LPI Feature

You can also conserve power by enabling the eco-friendly LPI (Low Power Idle) feature with the **ecofriendly lpi** command on page 10.14. This feature reduces the power supplied to the ports by the switch whenever the ports are idle and are connected to IEEE 802.3az Energy Efficient Ethernet compliant host devices. All ports configured for LPI must support LPI in hardware and must be configured to autonegotiate by default or by using the **speed** and **duplex** commands as needed.

LPI is a feature of the IEEE 802.3az Energy Efficient Ethernet (EEE) standard. LPI lowers power consumption of switch ports during periods of low link utilization when connected to IEEE 802.3az compliant host devices. If no data is sent then the switch port can enter a sleep state, called Low Power Idle (LPI), to conserve power used by the switch.

The eco-friendly LPI (Low Power Idle) feature is disabled by default. To enable the feature for a switch port, or for a range of switch ports in the example below, enter the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.6
awplus(config-if)# ecofriendly lpi
```

For an example of how to configure a trigger to enable the eco-friendly LPI feature, see **"Reduce Power Supplied to Ports" on page 102.9**.

Trouble-shoot Fiber and Pluggable Issues

Digital Diagnostics Monitoring (DDM) for SFP (1 Gigabit Small Form-factor Pluggable) and SFP+ (10 Gigabit Small Form-factor Pluggable) transceivers allows you to measure optical parameters for pluggables installed in a switch and trouble shoot fiber issues.

Fiber cable can be vulnerable to damage. Patch panels and patch cables can be connected with the wrong type of fiber, fiber splices can become faulty and fiber cables can be cut accidentally. Trouble shooting fiber issues has required special equipment and expertise to find the source of a problem causing signal attenuation. Digital monitoring features help find fiber issues.

Different types of transceivers are supported in different models of switch. See your Allied Telesis dealer for more information about the particular models of pluggables that your switch supports, and if these transceivers also support digital monitoring.

To display information about transceivers installed on your switch, enter the following command:

```
awplus# show system pluggable diagnostics
```

The following parameters are measured and are displayed in **show system pluggable diagnostics** command output:

- Temperature (Centigrade) inside the transceiver
- Vcc (Volts) voltage supplied to the transceiver
- Tx Bias (mA) current to the Laser Diode in the transceiver
- Tx Power (mW) the amount of light transmitted from the transceiver
- Rx Power (mW) the amount of light received in the transceiver

You can track Tx Bias to find out how the Laser Diode in the transceiver is aging by comparing the Tx Bias for one transceiver against Tx Bias for others. You can use this information to see if any transceivers may need replacement.

You can trouble shoot fiber connectivity issues by checking the Tx Power at one end of the fiber link against the Rx Power at the other end of the fiber link to measure the attenuation. Knowing the attenuation enables you to determine if there are anomalies in the fiber cable.

Note that Tx Power differences between the same type of transceivers installed on a switch may indicate that a transceiver is not seated or locked. Ensuring transceivers are seated and locked in place with the retaining clip will keep the fiber link up if there is any vibration or movement that can dislodge a fiber cable. Rx Power differences may indicate poor fiber patch cables, poor connectors or poor splices. Tracking Tx Bias for installed transceivers and measuring attenuation for fiber links allows you to perform periodic preventative maintenance, instead of reacting to a failure. Tracking Tx Power differences can be used as an indicator of failure in an which may need replacing.

Using the Find Me feature

The Find Me feature enables you to physically locate a specific device from a group of similar devices.

Running the **findme** command on page 10.16 causes the device's LEDs to alternately flash green and amber at a rate of 1 Hz. If the switch has no amber LED, then the green LED will flash on/off at a rate of 1 Hz.

An optional **interface** parameter specifies one or more interfaces to flash, while an optional **member** parameter specifies a particular stack member. Both these parameters are mutually exclusive. If no **interface** or **member** parameter is specified, then all ports on the device or stack are flashed.

An optional **timeout** parameter specifies the flash behavior **duration**. The default time is one minute (60 seconds). Normal LED behavior is restored automatically after either the default time, or a specified time, has elapsed or a **no findme** command is used.

Note At the time of writing, the AT-x510-28GPX and AT-x510-52GPX switches do not have the Find Me feature.

Continuous Reboot Prevention

Occasionally, due to network conditions or to recover from a software failure, the recovery mechanism of the switch is to reboot to resume normal operation. Provided the same error condition does not recur within a short period of time this is acceptable behavior. However, if the error condition repeatedly occurs within a short time period, the switch will go into a cycle of continuous reboots, causing network problems.

Although a switch continuously rebooting will come to the attention of a network administrator who can then resolve the issue, it is likely that in the meantime network problems have arisen. For example, a broadcast storm due to STP becoming unstable and trying to continually reconverge could cause the switch to reboot continuously.

In a VCStack situation, a continually rebooting switch will destabilize the stack and may cause the master and member devices to continually swap roles as they both reboot. This can seriously affect the network, because both devices will become too busy rebooting and forming the stack to forward traffic.

The continuous reboot prevention feature, enabled with the **continuous-reboot-prevention** command on page 10.11, allows the user to configure a switch to stop rebooting if the device gets into a cycle of continuous rebooting. The user can configure the time period, the maximum number of times the switch can reboot within the specified time period, referred to as the threshold, and the action to take if the threshold is exceeded.

There are three actions you can specify:

- linkdown

  The reboot procedure continues and all switch ports and stack ports stay link down. This is the default action.

- logonly

  The reboot procedure continues normally.

- stopreboot

  The reboot procedure stops and the user is prompted to enter the key "c" via the CLI. Normal reboot procedure then continues.

All reboot events are logged and the **show tech-support** command is executed automatically when continuous reboot prevention occurs.

Note that when the continuous reboot prevention feature is enabled on the switch, user initiated reboots via the CLI and software version auto-synchronization reboots (VCStack implementation) are not counted toward the threshold value.

The continuous reboot prevention feature is disabled by default. To enable the feature, enter the following commands:

```
awplus# configure terminal
awplus(config)# continuous-reboot-prevention enable
```

Unless the **period**, **threshold** and **action** parameter values are explicitly set, the defaults are used:

- period - 600 seconds
- threshold - 1 reboot event
- action - linkdown

To configure the **period**, **threshold** and the **action** to take if the number of reboots exceeds the specified threshold, enter the following commands:

```
awplus# configure terminal

awplus(config)# continuous-reboot-prevention [period
<60-604800>] [threshold <1-10>][action
[linkdown|logonly|stopreboot]]
```

If the action **stopreboot** is specified, the reboot procedure stops and the following message is displayed:

```
Please input key 'c' if you want to continue processing.
```

When the user has input “c” via the CLI, the reboot procedure continues.

To disable the continuous reboot prevention feature, enter the following commands:

```
awplus# configure terminal

awplus(config)# no continuous-reboot-prevention enable
```

To return either one or more of the **period**, **threshold** and the **action** parameters to the default, use the commands:

```
awplus# configure terminal

awplus(config)# no continuous-reboot-prevention [period]
[threshold] [action]
```

To display the current continuous reboot prevention configuration, enter the command:

```
awplus# show continuous-reboot-prevention
```

To display the reboot history of the switch, enter the command:

```
awplus# show reboot history
```

Controlling “show” Command Output

You can control the output of **show** commands by using the | and > or >> tokens in the following ways:

- To display only part of the output, follow the command with **|** and then other keywords (see **Output modifiers** below)
- To save the output to a file, follow the command with **> filename**
- To append the output to an existing file, follow the command with >> **filename**

Using the ? after typing the **show** command displays the following information about these tokens:

```
awplus#  show users
```

```
| Output modifiers
> Output redirection
>> Output redirection (append)
```

Output modifiers

Type the | (vertical bar) to use **output modifiers**.

```
append    Append output
begin     Begin with the first line that contains
          matching output
exclude   Exclude lines that contain matching output
include   Include lines that contain matching output
redirect  Redirect output
```

Begin

The **begin** parameter causes the display to begin at the first line that contains the input string.

```
awplus#  show run | begin vlan1
```

```
...skipping
interface vlan1
 ip address 192.168.14.1
!!
line con 0
 login
line vty 0 4
 login
!
end
```

Exclude The **exclude** parameter excludes all lines of output that contain the input string. In the following output all lines containing the word "input" are excluded:

```
awplus#  show interface vlan1 | exclude input
```

```
Interface vlan1
  Scope: both
  Hardware is Ethernet, address is 192.168.14.1
  index 3 metric 1 mtu 1500 <UP,BROADCAST,RUNNING,MULTICAST>
  Label switching is disabled
  No Virtual Circuit configured
  Administrative Group(s): None
  DSTE Bandwidth Constraint Mode is MAM
    output packets 4438, bytes 394940, dropped 0
    output errors 0, aborted 0, carrier 0, fifo 0, heartbeat 0,
window 0
    collisions 0
```

Include The include parameter includes only those lines of output that contain the input string. In the output below, all lines containing the word "input" are included:

```
awplus#  show interface vlan1 | include input
```

```
input packets 80434552, bytes 2147483647, dropped 0, multicast packets 0
input errors 0, length 0, overrun 0, CRC 0, frame 0, fifo 1, missed 0
```

Redirect The **redirect** parameter puts the lines of output into the specified file. If the file already exists, the new output overwrites the file's contents; the new output is not appended to the existing file contents.

| **redirect** and > are synonyms.

```
awplus#  show history | redirect history.txt
```

Output redirection The output redirection token > puts the lines of output into the specified file. If the file already exists, the new output overwrites the file's contents; the new output is not appended to the existing file contents.

| **redirect** and > are synonyms.

```
awplus#  show history > history.txt
```

Append output The append output token >> adds the lines of output into the specified file. The file must already exist, for the new output to be added to the end of the file's contents; the new output is appended to the existing file contents.

| **append** and >> are synonyms.

```
awplus#  show history >> history.txt
```

AlliedWare Plus GUI

Information on loading and using the AlliedWare Plus™ GUI is outside the scope of the main body of this reference manual. This topic is covered in a separate appendix to this document. See **"Appendix C: GUI Reference"**.

Chapter 2: Command Syntax Conventions in this Software Reference

The following table describes how command line interface syntax is shown in this Software Reference.

Syntax element	Example	What to enter in the command line

Keywords are shown in lowercase fixed-width font or bold variable-width font.	`show spanning-tree mst`
	or
	`show ip route`
		Some keywords are required, and others are optional parameters. Type keywords exactly as they appear in the command syntax.

Number ranges are enclosed in angle-brackets < > and separated by a hyphen.	*<0-255>*	Enter a number from the range. Do not enter the angle brackets.
Placeholders are shown in lowercase italics within angle-brackets < >, or in uppercase italics.	*<port-list>* or `ip dhcp pool` *NAME*	Replace the placeholder with the value you require. The placeholder may be an IP address, a text string, or some other value. See the parameter table for the command for information about the type of value to enter. Do not enter the angle-brackets.
Repeats are shown with ellipsis.	`param1...`	Enter the parameter one or more times.
Optional elements are shown in brackets: [ ]	`vlan` *<vid>* `[name` *<vlan-name>*`]`	If you need the optional parameter, enter it. Do not enter the brackets.
Required choices are enclosed in braces and separated by a vertical bar (pipe) : {|}	`spanning-tree {mstp|rstp|stp} enable`	Enter one only of the options. Do not enter the braces or vertical bar.
Optional choices are enclosed in or brackets and separated by a vertical bar (pipe): [|]	`[param1|param2]`	If needed, enter one only of the options. Do not enter the brackets or vertical bar.
Inclusive options are enclosed in braces, and separated by brackets: {[ ] [ ]}	`{[param1] [param2] [param3]}`	Enter one or more of the options and separate them with a space. Do not enter the braces or brackets.

Chapter 3: Start-Up Sequence

AlliedWare Plus Start-Up 3.2
Diagnostic Menu 3.3
Bootloader Menu 3.4
Start-Up Sequence 3.9

AlliedWare Plus Start-Up

Every switch has a start-up process. A specified version of product software must be loaded and executed. The bootloader is the executable code responsible for setting up the system and loading the release software.

The bootloader is the software that runs the unit when it first powers up, performing basic initialization and executing the product software release. As part of the start-up process of the switch, the bootloader allows you various options before running the product release software.

Previous versions of AlliedWare provide the option to boot to EPROM if a software release cannot be loaded, is unlicensed, or if selected by the user. The EPROM provides enough basic functionality to get a working software release loaded and operational on the switch. In AlliedWare Plus™ this task is handled by the bootloader.

As AlliedWare Plus™ begins its start-up process; there are two options that allow you to access either the diagnostic menu, or the bootloader menu. The following prompt is displayed when these options are temporarily available:

```
Bootloader 3.0.0-devel loaded
Press <Ctrl+B> for the Boot Menu
```

You can now enter one of the following two options to determine how the start-up process proceeds:

- Enter Ctrl+D to display the diagnostic menu.
- Enter Ctrl+B to display the bootloader menu.

Diagnostic Menu

Enter Ctrl+D during start-up to access the bootloader diagnostic menu, and provide options for performing various hardware tests. This can be useful as a tool for confirming a suspected hardware problem at the direction of network engineering personnel. When you enter Ctrl+D, the stage 1 diagnostics menu is displayed:

```
Bootup Stage 1 Diagnostics Menu:
  0. Restart
  1. Full RAM test
  2. Quick RAM test
  3. Battery backed RAM (NVS) test
  4. Bootloader ROM checksum test
  ---------------------------------
  7. Bootup stage 2 diagnostics menu
  ---------------------------------
  8. Quit to U-Boot shell
  9. Quit and continue booting
Enter selection ==>
```

The options in the stage 1 diagnostics menu allow you to initiate the following tests:

- **RAM:** The Bootloader fully tests any/all SDRAM installed in the system.
- **NVS:** The Bootloader fully tests any/all non-volatile (battery backed) SRAM installed in the system.
- **checksum:** The Bootloader checksum ROM memory for error detection.

For example, enter “2” to select a Quick RAM test:

```
Quick RAM test - press Q to quit, S to skip when failing
Writing pattern  ..............................
Checking pattern ..............................
Writing complemented pattern ..................
Checking complemented pattern ..................
Pass 1  total errors 0
```

Enter “7” to display the stage 2 diagnostics menu:

```
Entering stage 2...
Bootup Stage 2 Diagnostics Menu:
  0. Restart
  2. Test FLASH (Filesystem only)
  4. Erase FLASH (Filesystem only)
  5. Card slot test
  ---------------------------------
  8. Quit to U-Boot shell
  9. Quit and continue booting
```

The options in the stage 2 diagnostics menu allow you to initiate the following tests:

- **Flash:** The Bootloader tests the user file system area of Flash. The bootloader is stored in a protected area of Flash that is not accessed by the user file system.
- **Flash Erase:** The Bootloader erases the user file system area of Flash only.

Once any required tests are completed from the diagnostics menu, enter “9” to quit the diagnostic menu and continue the switch boot-up process.

Bootloader Menu

Enter Ctrl+B during start-up to access the bootloader menu where boot options can be set. The boot options shown are explained in detail under this example.

```
Boot Menu:

  ------------------------------------------------------
  B. Boot backup software
  ------------------------------------------------------
  0. Restart
  1. Perform one-off boot from alternate source
  2. Change the default boot source (for advanced users)
  3. Update Bootloader
  4. Adjust the console baud rate
  5. Special boot options
  6. System information
  7. Restore Bootloader factory settings
  ------------------------------------------------------
  9. Quit and continue booting
```

Boot options

A powerful feature of AlliedWare Plus™ is the ability to boot from a variety of sources. Previously the switch was constrained to just booting off the release loaded into Flash memory. The only software release upgrade path being to load a new release into Flash memory and then set this release to be loaded at the next restart.

With AlliedWare Plus™ the switch can boot from other sources, such as a USB stick or network server. This provides a very flexible system, with multiple options to upgrade software releases and for system recovery.

Details of the bootloader menu options are as follows:

1. Perform one-off boot from alternate source

Enter "1" in the top-level menu to provide the following one-off boot options:

```
Enter selection ==> 1

Select device:

  0. Return to previous menu
  ------------------------------------------------------
  1. Flash    (flash:)
  2. TFTP     (tftp://)
  3. YMODEM   (ymodem:)
  4. USB      (usb:)

Enter selection ==>
```

You can select a one-off boot from Flash, USB storage device, network server (TFTP), or ymodem. The selected option will be used for the next restart (only) of the switch. If you select to boot from the network, the bootloader prompts the user for the required network address details.

Note These settings are specific to the Bootloader. They are not related in any way to what may be configured by the main software release.

```
Enter IP version [4|6]......... [4]:
  Enter IP address for eth0 ..... [0.0.0.0]: 172.28.8.210
  Enter subnet mask ............. [0.0.0.0]: 255.255.0.0
  Enter gateway IP .............. [0.0.0.0]: 172.28.0.1
  Enter TFTP server IP .......... [0.0.0.0]: 172.30.1.242
  Enter filename ................ []: x510-5.4.4-3.5.rel
Loading tftp://172.30.1.242/x510-5.4.4-3.5.rel...
```

When the switch is booted up using the 'one-off' selected source for the software release, it provides the option to copy the release just used to Flash for further/permanent use:

```
login: manager
Password: ******
The system has been booted using the one off boot/recovery
mechanism.
Bootup has successfully completed.
Write this release to flash? (y/n):
```

2. Change the default boot source (for advanced users)

```
Boot Menu:

  ------------------------------------------------------
  B. Boot backup software
  ------------------------------------------------------
  0. Restart
  1. Perform one-off boot from alternate source
  2. Change the default boot source (for advanced users)
  3. Update Bootloader
  4. Adjust the console baud rate
  5. Special boot options
  6. System information
  7. Restore Bootloader factory settings
  ------------------------------------------------------
  9. Quit and continue booting
```

Entering “2” in the top-level menu provides the option to set the boot source permanently.

```
Enter selection ==> 2

NOTE: These settings are specific to the Bootloader.
  They are not related in any way to what may be configured
  by the 'boot system' command in the main software release.

Select device:

  0. Return to previous menu
  ------------------------------------------------------
  1. Flash    (flash:)
  2. TFTP     (tftp://)
  3. YMODEM   (ymodem:)
  4. USB      (usb:)
  ------------------------------------------------------
  9. Boot from default (determined by main CLI)

Enter selection ==>
```

The same boot source options are provided as with the one-off selection, but this time every restart of the switch will result in the unit booting from the selected source.

3. Update Bootloader

This option allows for the bootloader code to be updated. It is not detailed here, as it is envisioned that this would rarely need to be done, and only at the request of (and with support from) Allied Telesis engineering.

4. Adjust the console baud rate

```
Boot Menu:

  ------------------------------------------------------
  B. Boot backup software
  ------------------------------------------------------
  0. Restart
  1. Perform one-off boot from alternate source
  2. Change the default boot source (for advanced users)
  3. Update Bootloader
  4. Adjust the console baud rate
  5. Special boot options
  6. System information
  7. Restore Bootloader factory settings
  ------------------------------------------------------
  9. Quit and continue booting
```

The baud rate of the console session is set here to match the terminal program being used for management of the switch when connected directly to the asynchronous port. The switch's default value is 9600. The baud rate selected can be set as the 'new' default for future use if preferred.

```
Select baud rate:

  0. Return to previous menu
  ------------------------------------------------------
  1. 9600
  2. 19200
  3. 38400
  4. 57600
  5. 115200
  6. 230400 (Setting can't be made permanent)
  7. 460800 (Setting can't be made permanent)

Enter selection ==>  1

Change your terminal program baud rate to 9600 and press
enter...  if for some reason you are unable to do this,
power cycle the device and the existing baud rate will be
restored.
Use this baud rate by default? (Y/N) ==> n
```

5. Special boot options

```
Boot Menu:

  ------------------------------------------------------
  B. Boot backup software
  ------------------------------------------------------
  0. Restart
  1. Perform one-off boot from alternate source
  2. Change the default boot source (for advanced users)
  3. Update Bootloader
  4. Adjust the console baud rate
  5. Special boot options
  6. System information
  7. Restore Bootloader factory settings
  ------------------------------------------------------
  9. Quit and continue booting
```

The special boot options allow for system recovery in the event of a forgotten password or to the default configuration.

```
Special boot options menu:

  0. Return to previous menu
  ------------------------------------------------------
  1. Skip startup script (Use system defaults)

Enter selection ==>
```

6. System information

```
Boot Menu:

  ------------------------------------------------------
  B. Boot backup software
  ------------------------------------------------------
  0. Restart
  1. Perform one-off boot from alternate source
  2. Change the default boot source (for advanced users)
  3. Update Bootloader
  4. Adjust the console baud rate
  5. Special boot options
  6. System information
  7. Restore Bootloader factory settings
  ------------------------------------------------------
  9. Quit and continue booting
```

The system information option provides some details on the hardware platform in use, such as CPU, memory, hardware (MAC) address and so on.

7. Restore Bootloader factory settings

This option allows the bootloader to be set back to factory defaults.

Caution

This option erases any settings that may have been configured by this menu
Are you sure? (Y/N) ==>

The bootloader menu provides a powerful set of options for flexibility in the way software releases are upgraded on the switch, and system recovery is performed.

Start-Up Sequence

The start-up sequence for a device running AlliedWare Plus™ under normal circumstances will be as seen in the following pages. This sequence will be seen when everything loads and runs as expected.

Note

To enter the bootloader or diagnostic menus discussed previously, Ctrl+B or Ctrl+D must be entered when prompted before the software modules start loading.

```
Reading flash:x510-5.4.4-3.5.rel...

Verifying release... OK
Booting...
Starting base/first...                                  [  OK  ]
Mounting virtual filesystems...                         [  OK  ]
           _______________   ____
        /\ \             / /______\
      /   \ \_      __/ /| ______ |
    /      \ |     |  /  | ______ |
  /         \ \   / /     \ ____ /
/______/\____\ \/ /____________/
Allied Telesis Inc.
AlliedWare Plus (TM) vVersion 5.4.4-2
Current release filename: x510-5.4.4-3.5.rel
Original release filename: x510-5.4.4-3.5.rel
Built: Fri Feb 28 10:34:21 NZDT 2014
Mounting static filesystems...                          [  OK  ]
Checking flash filesystem...                            [  OK  ]
Mounting flash filesystem...                            [  OK  ]
Checking NVS filesystem...                              [  OK  ]
Mounting NVS filesystem...                              [  OK  ]
Starting base/dbus...                                   [  OK  ]
Starting base/syslog...                                 [  OK  ]
Starting base/loopback...                               [  OK  ]
Starting base/poe_done...                               [  OK  ]
Starting base/sysctl...                                 [  OK  ]
Starting base/portmapper...                             [  OK  ]
Received event syslog.done
Starting base/reboot-stability...                       [  OK  ]
Checking system reboot stability...                     [  OK  ]
Starting base/cron...                                   [  OK  ]
Starting base/appmond...                                [  OK  ]
Starting hardware/openhpi...                            [  OK  ]
Starting hardware/timeout...                            [  OK  ]
Starting base/inet...                                   [  OK  ]
Starting base/modules...                                [  OK  ]
Received event modules.done
Received event board.inserted
Starting network/poefw...                               [  OK  ]
Received event poefw.done
Received event hardware.done
Starting network/startup...                             [  OK  ]
Starting base/external-media...                         [  OK  ]
Starting network/stackd...                              [  OK  ]
Starting network/election.timeout...                    [  OK  ]
Received event network.enabled

Initializing HA processes:
hostd, cntrd, nsm, ripngd, sflowd, auth, epsr
hsl, imiproxyd, irdpd, lldpd, loopprot, mstp, ospf6d
pdmd, pim6d, ripd, rmon, vrrpd, atmfd, bgpd
lacp, ospfd, pimd, udldd, imi
Received event network.initialized
21:14:13 awplus-1 VCS[951]: No neighboring members found, unit may be in a stand
alone configuration
Received event vcs.elected-master
21:14:13 awplus-1 VCS[951]: Startup speed can be improved by adding 'no stack 1
enable' to configuration
21:14:13 awplus-1 VCS[951]: Member 1 (eccd.6d9d.4eed) has become the Active Mast
er
Assigning Active Workload to HA processes:
21:14:13 awplus VCS[951]: Stack Virtual MAC is 0000.cd37.0065
21:14:22 awplus ATMF[896]: host_eccd_6d9d_4eed has joined. 1 member in total.
authd, epsrd, hsl, irdpd, lacpd, lldpd, loopprotd
mstpd, nsm, rmond, sflowd, vrrpd, imi, imiproxyd
Received event network.activated

Loading configuration file flash:/default.cfg, please wait.
...
done!
Received event network.configured

awplus login: manager
Password:
```

There are three possible status results displayed for each module loaded - OK, INFO, ERROR:

- OK means that the module has loaded correctly.
- INFO means that an error occurred, but the device is usable.
- ERROR means that an error occurred and device operation may be affected.

Additional specific information accompanies an INFO or ERROR status result. For example, if a corrupt release file was set as the startup release, the following error message would be seen:

```
Bootloader 3.0.0-devel loaded
Press <Ctrl+B> for the Boot Menu
Reading filesystem...
Error: Release filename is invalid (should be x510-5.4.4-3.5.rel)
Error: There is no backup release file set
Error: Boot failed. Please recover the system using the Boot Menu
Restarting...Bootloader 3.0.0-devel loaded
Press <Ctrl+B> for the Boot Menu
```

Whether an error message results in a case of the device being unusable will depend on the specific error and message, so will need to be dealt with on a case by case basis. If a software release has been corrupted, as shown on start-up, a new release may need to be loaded.

Chapter 4: CLI Navigation Commands

Command List 4.2
configure terminal 4.2
disable (Privileged Exec mode) 4.2
do 4.3
enable (Privileged Exec mode) 4.4
end 4.6
exit 4.6
help 4.7
logout 4.7
show history 4.8

Command List

This chapter provides an alphabetical reference for the commands used to navigate between different modes. This chapter also provides a reference for the help and show commands used to help navigate within the CLI.

configure terminal

This command enters the Global Configuration command mode.

Syntax `configure terminal`

Mode Privileged Exec

Example To enter the Global Configuration command mode (note the change in the command prompt), enter the command:

```
awplus# configure terminal
awplus(config)#
```

disable (Privileged Exec mode)

This command exits the Privileged Exec mode, returning the prompt to the User Exec mode. To end a session, use the **exit** command.

Syntax `disable`

Mode Privileged Exec

Example To exit the Privileged Exec mode, enter the command:

```
awplus# disable
awplus>
```

Related Commands **enable (Privileged Exec mode)**
end
exit

do

This command lets you to run User Exec and Privileged Exec mode commands when you are in any configuration mode.

Syntax `do <command>`

Parameter	Description
<command>	Specify the command and its parameters.

Mode Any configuration mode

Example

```
awplus# configure terminal
awplus(config)# do ping 192.0.2.23
```

enable (Privileged Exec mode)

This command enters the Privileged Exec mode and optionally changes the privilege level for a session. If a privilege level is not specified then the maximum privilege level (15) is applied to the session. If the optional privilege level is omitted then only users with the maximum privilege level can access Privileged Exec mode without providing the password as specified by the **enable password** or **enable secret** commands. If no password is specified then only users with the maximum privilege level set with the **username** command can assess Privileged Exec mode.

Syntax `enable [<privilege-level>]`

Parameter	Description
<privilege-level>	Specify the privilege level for a CLI session in the range <1-15>, where 15 is the maximum privilege level, 7 is the intermediate privilege level and 1 is the minimum privilege level. The privilege level for a user must match or exceed the privilege level set for the CLI session for the user to access Privileged Exec mode. Privilege level for a user is configured by **username**.

Mode User Exec

Usage Many commands are available from the Privileged Exec mode that configure operating parameters for the switch, so you should apply password protection to the Privileged Exec mode to prevent unauthorized use. Passwords can be encrypted but then cannot be recovered. Note that non-encrypted passwords are shown in plain text in configurations.

The **username** command sets the privilege level for the user. After login, users are given access to privilege level 1. Users access higher privilege levels with the **enable (Privileged Exec mode)** command. If the privilege level specified is higher than the users configured privilege level specified by the **username** command, then the user is prompted for the password for that level.

Note that a separate password can be configured for each privilege level using the **enable password** and the **enable secret** commands from the Global Configuration mode. The **service password-encryption** command encrypts passwords configured by the **enable password** and the **enable secret** commands, so passwords are not shown in plain text in configurations.

Example The following example shows the use of the **enable** command to enter the Privileged Exec mode (note the change in the command prompt).

```
awplus> enable

awplus#
```

The following example shows the **enable** command enabling access the Privileged Exec mode for users with a privilege level of 7 or greater. Users with a privilege level of 7 or greater do not need to enter a password to access Privileged Exec mode. Users with a privilege level 6 or less need to enter a password to access Privilege Exec mode. Use the **enable password** command or the **enable secret** commands to set the password to enable access to Privileged Exec mode.

```
awplus> enable 7

awplus#
```

Related Commands

disable (Privileged Exec mode)
enable password
enable secret
exit
service password-encryption
username

end

This command returns the prompt to the Privileged Exec command mode from any other advanced command mode.

Syntax `end`

Mode All advanced command modes, including Global Configuration and Interface Configuration modes.

Example The following example shows the use of the `end` command to return to the Privileged Exec mode directly from Interface mode.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# end
awplus#
```

Related Commands **disable (Privileged Exec mode)**
enable (Privileged Exec mode)
exit

exit

This command exits the current mode, and returns the prompt to the mode at the previous level. When used in User Exec mode, the **exit** command terminates the session.

Syntax `exit`

Mode All command modes, including Global Configuration and Interface Configuration modes.

Example The following example shows the use of `exit` command to exit Interface mode, and return to Configure mode.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# exit
awplus(config)#
```

Related Commands **disable (Privileged Exec mode)**
enable (Privileged Exec mode)
end

help

This command displays a description of the AlliedWare Plus™ OS help system.

Syntax `help`

Mode All command modes

Example To display a description on how to use the system help, use the command:

```
awplus# help
```

Output **Figure 4-1: Example output from the help command**

```
When you need help at the command line, press '?'.

If nothing matches, the help list will be empty. Delete
characters until entering a '?' shows the available options.

Enter '?' after a complete parameter to show remaining valid
command parameters (e.g. 'show ?').

Enter '?' after part of a parameter to show parameters that
complete the typed letters (e.g. 'show ip?').
```

logout

This command exits the User Exec or Privileged Exec modes and ends the session.

Syntax `logout`

Mode User Exec and Privileged Exec

Example To exit the User Exec mode, use the command:

```
awplus# logout
```

show history

This command lists the commands entered in the current session. The history buffer is cleared automatically upon reboot.

The output lists all command line entries, including commands that returned an error.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show history`

Mode User Exec and Privileged Exec

Example To display the commands entered during the current session, use the command:

```
awplus# show history
```

Output **Figure 4-2: Example output from the show history command**

```
    1 en
    2 show ru
    3 con t
    4 route-map er deny 3
    5 exit
    6 ex
    7 di
```

Chapter 5: User Access Commands

Introduction 5.2
Command List 5.2
clear line console 5.2
clear line vty 5.3
enable password 5.4
  Using plain passwords 5.5
  Using encrypted passwords 5.5
  Using hidden passwords 5.6
enable secret 5.7
  Using plain passwords 5.8
  Using encrypted passwords 5.8
  Using hidden passwords 5.9
exec-timeout 5.10
flowcontrol hardware (asyn/console) 5.11
length (asyn) 5.13
line 5.14
privilege level 5.16
security-password history 5.17
security-password forced-change 5.18
security-password lifetime 5.19
security-password minimum-categories 5.20
security-password minimum-length 5.21
security-password reject-expired-pwd 5.22
security-password warning 5.23
service advanced-vty 5.24
service http 5.25
service password-encryption 5.26
service telnet 5.27
service terminal-length 5.28
show privilege 5.29
show security-password configuration 5.30
show security-password user 5.31
show telnet 5.32
show users 5.33
telnet 5.34
telnet server 5.35
terminal length 5.36
terminal resize 5.37
username 5.38

Introduction

This chapter provides an alphabetical reference of commands used to configure user access.

Command List

clear line console

This command resets a console line. If a terminal session exists on the line then the terminal session is terminated. If console line settings have changed then the new settings are applied.

Syntax `clear line console 0`

Mode Privileged Exec

Example To reset the console line (asyn), use the command:

```
awplus# clear line console 0

        % The new settings for console line 0 have been
        applied
```

Related Commands
clear line vty
flowcontrol hardware (asyn/console)
line
show users

clear line vty

This command resets a VTY line. If a session exists on the line then it is closed.

Syntax `clear line vty <0-32>`

Parameter	Description
<0-32>	Line number

Mode Privileged Exec

Example To reset the first vty line, use the command:

```
awplus# clear line vty 1
```

Related Commands
privilege level
line
show telnet
show users

enable password

To set a local password to control access to various privilege levels, use the **enable password** Global Configuration command. Use the **enable password** command to modify or create a password to be used, and use the **no enable password** command to remove the password.

Note that the **enable secret** command is an alias for the **enable password** command, and the **no enable secret** command is an alias for the **no enable password** command. Issuing a **no enable password** command removes a password configured with the **enable secret** command. The **enable password** command is shown in the running and startup configurations. Note that if the **enable secret** command is entered then **enable password** is shown in the configuration.

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option `8` for GUI users.

Syntax

```
enable password [<plain>|8 <hidden>|level <1-15> 8 <hidden>]
no enable password [level <1-15>]
```

Parameter	Description
<plain>	Specifies the unencrypted password.
`8`	Specifies a hidden password will follow.
<hidden>	Specifies the hidden encrypted password. Use an encrypted password for better security where a password crosses the network or is stored on a TFTP server.
`level`	Privilege level <1-15>. Level for which the password applies. You can specify up to 16 privilege levels, using numbers 1 through 15. Level 1 is normal EXEC-mode user privileges for **User Exec** mode. If this argument is not specified in the command or the **no** variant of the command, the privilege level defaults to 15 (enable mode privileges) for **Privileged Exec** mode. A privilege level of 7 can be set for intermediate CLI security.

Default The privilege level for enable password is level 15 by default. Previously the default was level 1.

Mode Global Configuration

Usage This command enables the Network Administrator to set a password for entering the Privileged Exec mode when using the **enable (Privileged Exec mode)** command. There are three methods to enable a password. In the examples below, for each method, note that the configuration is different and the configuration file output is different, but the password string to be used to enter the Privileged Exec mode with the **enable** command is the same (**mypasswd**).

A user can now have an intermediate CLI security level set with this command for privilege level 7 to access all the show commands in Privileged Exec mode and all the commands in User Exec mode, but not any configuration commands in Privileged Exec mode.

Note that the **enable password** command is an alias for the **enable secret** command and one password per privilege level is allowed using these commands. Do not assign one password to a privilege level with **enable password** and another password to a privilege level with **enable secret**. Use **enable password** or **enable secret** commands. Do not use both on the same level.

Using plain passwords

The plain password is a clear text string that appears in the configuration file as configured.

```
awplus#  configure terminal
awplus(config)#  enable password mypasswd
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password mypasswd
!
interface lo
```

Using encrypted passwords

Configure an encrypted password using the **service password-encryption** command. First, use the enable password command to specify the string that you want to use as a password (**mypasswd**). Then, use the **service password-encryption** command to encrypt the specified string (**mypasswd**). The advantage of using an encrypted password is that the configuration file does not show **mypasswd**, it will only show the encrypted string **fU7zHzuutY2SA**.

Note

Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option `8` for GUI users.

```
awplus#  configure terminal
awplus(config)#  enable password mypasswd
awplus(config)#  service password-encryption
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
service password-encryption
!
interface lo
```

Using hidden passwords

Configure an encrypted password using the **HIDDEN** parameter (**8**) with the **enable password** command. Use this method if you already know the encrypted string corresponding to the plain text string that you want to use as a password. It is not required to use the **service password-encryption** command for this method. The output in the configuration file will show only the encrypted string, and not the text string.

```
awplus#  configure terminal
awplus(config)#  enable password 8 fU7zHzuutY2SA
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
!
interface lo
```

Related Commands

enable (Privileged Exec mode)
enable secret
service password-encryption
privilege level
show privilege
username
show running-config

enable secret

To set a local password to control access to various privilege levels, use the **enable secret** Global Configuration command. Use the **enable secret** command to modify or create a password to be used, and use the **no enable secret** command to remove the password.

Note that the **enable secret** command is an alias for the **enable password** command, and the **no enable secret** command is an alias for the **no enable password** command. Issuing a **no enable password** command removes a password configured with the **enable secret** command. The **enable password** command is shown in the running and startup configurations. Note that if the **enable secret** command is entered then **enable password** is shown in the configuration.

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option `8` for GUI users.

Syntax

```
enable secret [<plain>|8 <hidden>|level <0-15> 8 <hidden>]
no enable secret [level <1-15>]
```

Parameter	Description
<plain>	Specifies the unencrypted password.
`8`	Specifies a hidden password will follow.
<hidden>	Specifies the hidden encrypted password. Use an encrypted password for better security where a password crosses the network or is stored on a TFTP server.
`level`	Privilege level <1-15>. Level for which the password applies. You can specify up to 16 privilege levels, using numbers 1 through 15. Level 1 is normal EXEC-mode user privileges for **User Exec** mode. If this argument is not specified in the command or the **no** variant of the command, the privilege level defaults to 15 (enable mode privileges) for **Privileged Exec** mode. A privilege level of 7 can be set for intermediate CLI security.

Default The privilege level for enable secret is level 15 by default.

Mode Global Configuration

Usage This command enables the Network Administrator to set a password for entering the Privileged Exec mode when using the **enable (Privileged Exec mode)** command. There are three methods to enable a password. In the examples below, for each method, note that the configuration is different and the configuration file output is different, but the password string to be used to enter the Privileged Exec mode with the **enable** command is the same (**mypasswd**).

A user can have an intermediate CLI security level set with this command for privilege level 7 to access all the show commands in Privileged Exec mode and all the commands in User Exec mode, but not any configuration commands in Privileged Exec mode.

Note that the **enable secret** command is an alias for the **enable password** command and one password per privilege level is allowed using these commands. Do not assign one password to a privilege level with **enable password** and another password to a privilege level with **enable secret**. Use **enable password** or **enable secret** commands. Do not use both on the same level.

Using plain passwords

The plain password is a clear text string that appears in the configuration file as configured.

```
awplus#  configure terminal
awplus(config)#  enable secret mypasswd
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password mypasswd
!
interface lo
```

Using encrypted passwords

Configure an encrypted password using the **service password-encryption** command. First, use the enable password command to specify the string that you want to use as a password (**mypasswd**). Then, use the **service password-encryption** command to encrypt the specified string (**mypasswd**). The advantage of using an encrypted password is that the configuration file does not show **mypasswd**, it will only show the encrypted string **fU7zHzuutY2SA**.

Note

Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option `8` for GUI users.

```
awplus#  configure terminal
awplus(config)#  enable secret mypasswd
awplus(config)#  service password-encryption
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
service password-encryption
!
interface lo
```

Using hidden passwords

Configure an encrypted password using the **HIDDEN** parameter (**8**) with the **enable password** command. Use this method if you already know the encrypted string corresponding to the plain text string that you want to use as a password. It is not required to use the **service password-encryption** command for this method. The output in the configuration file will show only the encrypted string, and not the text string:

```
awplus#  configure terminal
awplus(config)#  enable secret 8 fU7zHzuutY2SA
awplus(config)#  end
```

This results in the following show output:

```
awplus#show run
Current configuration:
hostname awplus
enable password 8 fU7zHzuutY2SA
!
interface lo
```

Related Commands

enable (Privileged Exec mode)
enable secret
service password-encryption
privilege level
show privilege
username
show running-config

exec-timeout

This command sets the interval your device waits for user input from either a console or VTY connection. Once the timeout interval is reached, the connection is dropped. This command sets the time limit when the console or VTY connection automatically logs off after no activity.

The **no** variant of this command removes a specified timeout and resets to the default timeout (10 minutes).

Syntax

```
exec-timeout {<minutes>} [<seconds>]
no exec-timeout
```

Parameter	Description
<minutes>	<0-35791> Required integer timeout value in minutes
<seconds>	<0-2147483> Optional integer timeout value in seconds

Default The default for the **exec-timeout** command is 10 minutes and 0 seconds (**exec-timeout 10 0**).

Mode Line Configuration

Usage This command is used set the time the telnet session waits for an idle VTY session, before it times out. An **exec-timeout 0 0** setting will cause the telnet session to wait indefinitely. The command **exec-timeout 0 0** is useful while configuring a device, but reduces device security.

If no input is detected during the interval then the current connection resumes. If no connections exist then the terminal returns to an idle state and disconnects incoming sessions.

Examples To set VTY connections to timeout after 2 minutes, 30 seconds if there is no response from the user, use the following commands:

```
awplus# configure terminal
awplus(config)# line vty 0 32
awplus(config-line)# exec-timeout 2 30
```

To reset the console connection to the default timeout of 10 minutes 0 seconds if there is no response from the user, use the following commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# no exec-timeout
```

Validation Commands **show running-config**

Related Commands **line**
service telnet

flowcontrol hardware (asyn/console)

Use this command to enable RTS/CTS (Ready To Send/Clear To Send) hardware flow control on a terminal console line (asyn port) between the DTE (Data Terminal Equipment) and the DCE (Data Communications Equipment).

Syntax

```
flowcontrol hardware
no flowcontrol hardware
```

Mode Line Configuration

Default Hardware flow control is disabled by default.

Usage Hardware flow control makes use of the RTS and CTS control signals between the DTE and DCE where the rate of transmitted data is faster than the rate of received data. Flow control is a technique for ensuring that a transmitting entity does not overwhelm a receiving entity with data. When the buffers on the receiving device are full, a message is sent to the sending device to suspend the transmission until the data in the buffers has been processed.

Hardware flow control can be configured on terminal console lines (e.g. asyn0). For Reverse Telnet connections, hardware flow control must be configured to match on both the Access Server and the Remote Device. For terminal console sessions, hardware flow control must be configured to match on both the DTE and the DCE. Settings are saved in the running configuration. Changes are applied after reboot, clear line console, or after closing the session.

Use **show running-config** and **show startup-config** commands to view hardware flow control settings that take effect after reboot for a terminal console line. See the **show running-config** command output:

```
awplus#show running-config
!
line con 1
  speed 9600
  mode out 2001
  flowcontrol hardware
!
```

Note that line configuration commands do not take effect immediately. Line configuration commands take effect after one of the following commands or events:

- issuing a **clear line console** command
- issuing a **reboot** command
- logging out of the current session

Examples To enable hardware flow control on terminal console line asyn0, use the commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# flowcontrol hardware
```

To disable hardware flow control on terminal console line asyn0, use the commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# no flowcontrol hardware
```

Related Commands

clear line console
show running-config
speed (asyn)

length (asyn)

Use this command to specify the number of rows of output that the device will display before pausing, for the console or VTY line that you are configuring.

The **no** variant of this command restores the length of a line (terminal session) attached to a console port or to a VTY to its default length of 22 rows.

Syntax

```
length <0-512>
no length
```

Parameter	Description
<0-512>	Number of lines on screen. Specify 0 for no pausing.

Mode Line Configuration

Default The length of a terminal session is 22 rows. The **no length** command restores the default.

Usage If the output from a command is longer than the length of the line the output will be paused and the '–More–' prompt allows you to move to the next screen full of data.

A length of 0 will turn off pausing and data will be displayed to the console as long as there is data to display.

Examples To set the terminal session length on the console to 10 rows, use the command:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# length 10
```

To reset the terminal session length on the console to the default (22 rows), use the command:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# no length
```

To display output to the console continuously, use the command:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# length 0
```

Related Commands
service terminal-length
terminal length
terminal resize

line

Use this command to enter line configuration mode for the specified VTYs or the console. The command prompt changes to show that the switch is in Line Configuration mode.

Syntax

```
line vty <first-line> [<last-line>]
line console 0
```

Parameter	Description
<first-line>	*<0-32>* Specify the first line number.
<last-line>	*<0-32>* Specify the last line number.
console	The console terminal line(s) for local access.
vty	Virtual terminal for remote console access.

Mode Global Configuration

Usage In Line Configuration mode, you can configure console and virtual terminal settings, including setting **speed (asyn)**, **length (asyn)**, **privilege level**, and authentication (**login authentication**) or accounting (**accounting login**) method lists.

To change the console (asyn) port speed, use this **line** command to enter Line Configuration mode before using the **speed (asyn)** command on page 10.66. Set the console speed (Baud rate) to match the transmission rate of the device connected to the console (asyn) port on your switch.

Note that line configuration commands do not take effect immediately. Line configuration commands take effect after one of the following commands or events:

- issuing a **clear line console** command
- issuing a **reboot** command
- logging out of the current session

Examples To enter Line Configuration mode in order to configure all VTYs, use the commands:

```
awplus# configure terminal
awplus(config)# line vty 0 32
awplus(config-line)#
```

To enter Line Configuration mode to configure the console (asyn 0) port terminal line, use the commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)#
```

Related Commands

accounting login
clear line console
clear line vty
flowcontrol hardware (asyn/console)
length (asyn)
login authentication
privilege level
speed (asyn)

privilege level

This command sets a privilege level for VTY or console connections. The configured privilege level from this command overrides a specific user's initial privilege level at the console login.

Syntax `privilege level <1-15>`

Mode Line Configuration

Usage You can set an intermediate CLI security level for a console user with this command by applying privilege level 7 to access all show commands in Privileged Exec and all User Exec commands. However, intermediate CLI security will not show configuration commands in Privileged Exec.

Examples To set the console connection to have the maximum privilege level, use the following commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# privilege level 15
```

To set all vty connections to have the minimum privilege level, use the following commands:

```
awplus# configure terminal
awplus(config)# line vty 0 5
awplus(config-line)# privilege level 1
```

To set all vty connections to have an intermediate CLI security level, to access all show commands, use the following commands:

```
awplus# configure terminal
awplus(config)# line vty 0 5
awplus(config-line)# privilege level 7
```

Related Commands **enable password**
line
show privilege
username

security-password history

This command specifies the number of previous passwords that are unable to be reused. A new password is invalid if it matches a password retained in the password history.

The **no security-password history** command disables the security password history functionality.

Syntax

```
security-password history <0-15>

no security-password history
```

Parameter	Description
<0-15>	The allowable range of previous passwords to match against. A value of *0* will disable the history functionality and is equivalent to the **no security-password history** command. If the history functionality is disabled, all users' password history is reset and all password history is lost.

Default The default history value is `0`, which will disable the history functionality.

Mode Global Configuration

Examples To restrict reuse of the three most recent passwords, use the command:

```
awplus# configure terminal
awplus(config)# security-password history 3
```

To allow the reuse of recent passwords, use the command:

```
awplus# configure terminal
awplus(config)# no security-password history
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd
security-password warning

security-password forced-change

This command specifies whether or not a user is forced to change an expired password at the next login. If this feature is enabled, users whose passwords have expired are forced to change to a password that must comply with the current password security rules at the next login.

Note that to use this command, the lifetime feature must be enabled with the **security-password lifetime** command and the reject-expired-pwd feature must be disabled with the **security-password reject-expired-pwd** command.

The **no security-password forced-change** command disables the forced-change feature.

Syntax

```
security-password forced-change
no security-password forced-change
```

Default The forced-change feature is disabled by default.

Mode Global Configuration

Example To force a user to change their expired password at the next login, use the command:

```
awplus# configure terminal
awplus(config)# security-password forced-change
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password history
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd
security-password warning

security-password lifetime

This command enables password expiry by specifying a password lifetime in days.

Note that when the password lifetime feature is disabled, it also disables the **security-password forced-change** command and the **security-password warning** command.

The **no security-password lifetime** command disables the password lifetime feature.

Syntax

```
security-password lifetime <0-1000>
no security-password lifetime
```

Parameter	Description
<0-1000>	Password lifetime specified in days. A value of *0* will disable lifetime functionality and the password will never expire. This is equivalent to the **no security-password lifetime** command.

Default The default password lifetime is `0`, which will disable the lifetime functionality.

Mode Global Configuration

Example To configure the password lifetime to 10 days, use the command:

```
awplus# configure terminal
awplus(config)# security-password lifetime 10
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password history
security-password forced-change
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd
security-password warning
show security-password user

security-password minimum-categories

This command specifies the minimum number of categories that the password must contain in order to be considered valid. The password categories are:

- uppercase letters: A to Z
- lowercase letters: a to z
- digits: 0 to 9
- special symbols: all printable ASCII characters not included in the previous three categories. The question mark (?) cannot be used as it is reserved for help functionality.

Note that to ensure password security, the minimum number of categories should align with the lifetime selected, i.e. the fewer categories specified the shorter the lifetime specified.

Syntax

```
security-password minimum-categories <1-4>
```

Parameter	Description
<1-4>	Number of categories the password must satisfy, in the range *1* to *4*.

Default The default number of categories that the password must satisfy is `1`.

Mode Global Configuration

Example To configure the required minimum number of character categories to be *3*, use the command:

```
awplus# configure terminal
awplus(config)# security-password minimum-categories 3
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password history
security-password forced-change
security-password lifetime
security-password minimum-length
security-password reject-expired-pwd
security-password warning
username

security-password minimum-length

This command specifies the minimum allowable password length. This value is checked against when there is a password change or a user account is created.

Syntax `security-password minimum-length <1-23>`

Parameter	Description
<1-23>	Minimum password length in the range from 1 to 23.

Default The default minimum password length is `1`.

Mode Global Configuration

Example To configure the required minimum password length as `8`, use the command:

```
awplus# configure terminal
awplus(config)# security-password minimum-length 8
```

Validation Commands **show running-config security-password**
show security-password configuration

Related Commands **security-password history**
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password reject-expired-pwd
security-password warning
username

security-password reject-expired-pwd

This command specifies whether or not a user is allowed to login with an expired password. Users with expired passwords are rejected at login if this functionality is enabled. Users then have to contact the Network Administrator to change their password.

Caution

Once all users' passwords are expired you are unable to login to the device again if the security-password reject-expired-pwd command has been executed. You will have to reboot the device with a default configuration file, or load an earlier software version that does not have the security password feature.
We recommend you never have the command line "security-password reject-expired-pwd" in a default config file.

Note that when the reject-expired-pwd functionality is disabled and a user logs on with an expired password, if the forced-change feature is enabled with **security-password forced-change** command, a user may have to change the password during login depending on the password lifetime specified by the **security-password lifetime** command.

The **no security-password reject-expired-pwd** command disables the reject-expired-pwd feature.

Syntax

```
security-password reject-expired-pwd
no security-password reject-expired-pwd
```

Default The reject-expired-pwd feature is disabled by default.

Mode Global Configuration

Example To configure the system to reject users with an expired password, use the command:

```
awplus# configure terminal
awplus(config)# security-password reject-expired-pwd
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password history
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password warning
show security-password user

security-password warning

This command specifies the number of days before the password expires that the user will receive a warning message specifying the remaining lifetime of the password.

Note that the warning period cannot be set unless the lifetime feature is enabled with the **security-password lifetime** command.

The **no security-password warning** command disables this feature.

Syntax

```
security-password warning <0-1000>
no security-password warning
```

Parameter	Description
<0-1000>	Warning period in the range from *0* to *1000* days. A value *0* disables the warning functionality and no warning message is displayed for expiring passwords. This is equivalent to the **no security-password warning** command. The warning period must be less than, or equal to, the password lifetime set with the **security-password lifetime** command.

Default The default warning period is `0`, which disables warning functionality.

Mode Global Configuration

Example To configure a warning period of three days, use the command:

```
awplus# configure terminal
awplus(config)# security-password warning 3
```

Validation Commands
show running-config security-password
show security-password configuration

Related Commands
security-password history
security-password forced-change
security-password lifetime
security-password minimum-categories
security-password minimum-length
security-password reject-expired-pwd

service advanced-vty

This command enables the advanced-vty help feature. This allows you to use TAB completion for commands. Where multiple options are possible, the help feature displays the possible options.

The **no service advanced-vty** command disables the advanced-vty help feature.

Syntax

```
service advanced-vty
no service advanced-vty
```

Default The advanced-vty help feature is enabled by default.

Mode Global Configuration

Examples To disable the advanced-vty help feature, use the command:

```
awplus# configure terminal
awplus(config)# no service advanced-vty
```

To re-enable the advanced-vty help feature after it has been disabled, use the following commands:

```
awplus# configure terminal
awplus(config)# service advanced-vty
```

service http

This command enables the HTTP (Hypertext Transfer Protocol) service. The HTTP service is enabled by default and is required to support the AlliedWare Plus™ GUI Java applet on a Java enabled browser. See **Appendix C: GUI Reference** for further information about installing and using the AlliedWare Plus™ GUI.

The **no service http** command disables the HTTP feature.

Syntax

```
service http
no service http
```

Default The HTTP service is enabled by default.

Mode Global Configuration

Examples To disable the HTTP service, use the command:

```
awplus# configure terminal
awplus(config)# no service http
```

To re-enable the HTTP service after it has been disabled, use the following commands:

```
awplus# configure terminal
awplus(config)# service http
```

service password-encryption

Use this command to enable password encryption. This is enabled by default. When password encryption is enabled, the device displays passwords in the running config in encrypted form instead of in plain text.

Use the **no service password-encryption** command to stop the device from displaying newly-entered passwords in encrypted form. This does not change the display of existing passwords.

Note Do not use encrypted passwords for GUI users. The GUI requires unencrypted user passwords only - not encrypted user passwords. Do not use option `8` for GUI users.

Syntax

```
service password-encryption
no service password-encryption
```

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# service password-encryption
```

Validation Commands **show running-config**

Related Commands **enable password**

service telnet

Use this command to enable the telnet server. The server is enabled by default. Enabling the telnet server starts the switch listening for incoming telnet sessions on the configured port.

The server listens on port 23, unless you have changed the port by using the **privilege level** command on page 5.16.

Use the **no** variant of this command to disable the telnet server. Disabling the telnet server will stop the switch listening for new incoming telnet sessions. However, existing telnet sessions will still be active.

Syntax

```
service telnet [ip|ipv6]
no service telnet [ip|ipv6]
```

Default The IPv4 and IPv6 telnet servers are enabled by default.

The configured telnet port is TCP port 23 by default.

Mode Global Configuration

Examples To enable both the IPv4 and IPv6 telnet servers, use the following commands:

```
awplus# configure terminal
awplus(config)# service telnet
```

To enable the IPv6 telnet server only, use the following commands:

```
awplus# configure terminal
awplus(config)# service telnet ipv6
```

To disable both the IPv4 and IPv6 telnet servers, use the following commands:

```
awplus# configure terminal
awplus(config)# no service telnet
```

To disable the IPv6 telnet server only, use the following commands:

```
awplus# configure terminal
awplus(config)# no service telnet ipv6
```

Related Commands

clear line vty
show telnet
telnet server

service terminal-length

Use this command to specify the number of rows of output that the device will display before pausing, for all console and VTY lines.

Use the **no** variant of this command to remove the length specified by this command. The default length will apply unless you have changed the length for some or all lines by using the **length (asyn)** command on page 5.13.

Syntax

```
service terminal-length <lines>
no service terminal-length <lines>
```

Parameter	Description
`terminal-length`	Establish system-wide terminal length configuration.
`<lines>`	`<0-512>` Number of rows that the device will display before pausing.

Mode Global Configuration

Usage This command overrides any lengths set by using the **length (asyn)** command on page 5.13 in Line mode.

Example To display 60 rows of text before pausing, use the following command:

```
awplus# configure terminal
awplus(config)# service terminal-length 60
```

Related Commands

service terminal-length
terminal length
terminal resize

show privilege

This command displays the current user privilege level, which can be any privilege level in the range <1-15>. Privilege levels <1-6> allow limited user access (all User Exec commands), privilege levels <7-14> allow restricted user access (all User Exec commands plus Privileged Exec show commands). Privilege level 15 gives full user access to all Privileged Exec commands.

Syntax `show privilege`

Mode User Exec and Privileged Exec

Usage A user can have an intermediate CLI security level set with this command for privilege levels <7-14> to access all show commands in Privileged Exec mode and all commands in User Exec mode, but no configuration commands in Privileged Exec mode.

Example To show the current privilege level of the user, use the command:

```
awplus# show privilege
```

Output **Figure 5-1: Example output from the show privilege command**

```
awplus#show privilege
Current privilege level is 15
awplus#disable
awplus>show privilege
Current privilege level is 1
```

Related Commands privilege level

show security-password configuration

This command displays the configuration settings for the various security password rules.

Syntax `show security-password configuration`

Mode Privileged Exec

Example To display the current security-password rule configuration settings, use the command:

```
awplus# show security-password configuration
```

Output **Figure 5-2: Example output from the show security-password configuration command**

```
Security Password Configuration
Minimum password length ............................ 8
Minimum password character categories to match ..... 3
Number of previously used passwords to restrict..... 4
Password lifetime .................................. 30 day(s)
  Warning period before password expires ........... 3 day(s)
Reject expired password at login ................... Disabled
  Force changing expired password at login ......... Enabled
```

Related Commands **show running-config security-password**
show security-password user

show security-password user

This command displays user account and password information for all users.

Syntax `show security-password user`

Mode Privileged Exec

Example To display the system users' remaining lifetime or last password change, use the command:

```
awplus# show security-password user
```

Output **Figure 5-3: Example output from the show security-password user command**

```
User account and password information

UserName          Privilege Last-PWD-Change   Remaining-lifetime
------------------------------------------------------------------
manager           15          4625 day(s) ago   No Expiry
bob15             15             0 day(s) ago   30 days
ted7              7              0 day(s) ago   No Expiry
mike1             1              0 day(s) ago   No Expiry
```

Related Commands **show running-config security-password**
show security-password configuration

show telnet

This command shows the Telnet server settings.

Syntax `show telnet`

Mode User Exec and Privileged Exec

Example To show the Telnet server settings, use the command:

```
awplus# show telnet
```

Output **Figure 5-4: Example output from the show telnet command**

```
Telnet Server Configuration
-------------------------------------------------------------
Telnet server               : Enabled
Protocol                    : IPv4,IPv6
Port                        : 23
```

Related Commands
clear line vty
service telnet
show users
telnet server

show users

This command shows information about the users who are currently logged into the device.

Syntax `show users`

Mode User Exec and Privileged Exec

Example To show the users currently connected to the device, use the command:

```
awplus# show users
```

Output **Figure 5-5: Example output from the show users command**

```
Line    User            Host(s)  Idle      Location        Priv Idletime Timeout
con 0   manager         idle     00:00:00  ttyS0           15   10       N/A
vty 0   bob             idle     00:00:03  172.16.11.3     1    0        5
```

Table 5-1: Parameters in the output of the show users command

Parameter	Description
`Line`	Console port user is connected to.
`User`	Login name of user.
`Host(s)`	Status of the host the user is connected to.
`Idle`	How long the host has been idle.
`Location`	URL location of user.
`Priv`	The privilege level in the range 1 to 15, with 15 being the highest.
`Idletime`	The time interval the device waits for user input from either a console or VTY connection.
`Timeout`	The time interval before a server is considered unreachable.

telnet

Use this command to open a telnet session to a remote device.

Syntax `telnet {<hostname>|[ip] <ipv4-addr>|[ipv6] <ipv6-addr>} [<port>]`

Parameter	Description
<hostname>	The host name of the remote system.
`ip`	Keyword used to specify the IPv4 address or host name of a remote system.
<ipv4-addr>	An IPv4 address of the remote system.
`ipv6`	Keyword used to specify the IPv6 address of a remote system
<ipv6-addr>	Placeholder for an IPv6 address in the format `x:x::x:x`, for example, `2001:db8::8a2e:7334`
<port>	Specify a TCP port number (well known ports are in the range 1-1023, registered ports are 1024-49151, and private ports are 49152-65535).

Mode User Exec and Privileged Exec

Examples To connect to TCP port `2602` on the device at `10.2.2.2`, use the command:

```
awplus# telnet 10.2.2.2 2602
```

To connect to the telnet server `host.example`, use the command:

```
awplus# telnet host.example
```

To connect to the telnet server `host.example` on TCP port `100`, use the command:

```
awplus# telnet host.example 100
```

telnet server

This command enables the telnet server on the specified TCP port. If the server is already enabled then it will be restarted on the new port. Changing the port number does not affect the port used by existing sessions.

Syntax `telnet server {<1-65535>|default}`

Parameter	Description
<1-65535>	The TCP port to listen on.
`default`	Use the default TCP port number 23.

Mode Global Configuration

Example To enable the telnet server on TCP port 2323, use the following commands:

```
awplus# configure terminal
awplus(config)# telnet server 2323
```

Related Commands show telnet

terminal length

Use the **terminal length** command to specify the number of rows of output that the device will display before pausing, for the currently-active terminal only.

Use the **terminal no length** command to remove the length specified by this command. The default length will apply unless you have changed the length for some or all lines by using the **length (asyn)** command on page 5.13.

Syntax

```
terminal length <length>
terminal no length [<length>]
```

Parameter	Description
<length>	<0-512> Number of rows that the device will display on the currently-active terminal before pausing.

Mode User Exec and Privileged Exec

Examples The following example sets the number of lines to 15:

```
awplus# terminal length 15
```

The following example removes terminal length set previously:

```
awplus# terminal no length
```

Related Commands

length (asyn)
service terminal-length
terminal resize

terminal resize

Use this command to automatically adjust the number of rows of output on the console, which the device will display before pausing, to the number of rows configured on the user's terminal.

Syntax `terminal resize`

Mode User Exec and Privileged Exec

Usage When the user's terminal size is changed, then a remote session via SSH or TELNET adjusts the terminal size automatically. However, this cannot normally be done automatically for a serial or console port. This command automatically adjusts the terminal size for a serial or console port.

Examples The following example automatically adjusts the number of rows shown on the console:

```
awplus# terminal resize
```

Related Commands **length (asyn)**
service terminal-length
terminal length

username

This command creates or modifies a user to assign a privilege level and a password.

Note The default username privilege level of 1 is not shown in running-config output. Any username privilege level that has been modified from the default is shown.

Syntax

```
username <name> privilege <0-15> [password [8] <password>]
username <name> password [8] <password>
no username <name>
```

Parameter	Description	
<name>	The login name for the user. Do not use punctuation marks such as single quotes (‘ ‘), double quotes (“ “), or colons (:) with the user login name.	
`privilege`	The user’s privilege level. Use the privilege levels to set the access rights for each user.	
	<0-15>	A privilege level: either 0 (no access), 1-14 (limited access) or 15 (full access). A user with privilege level 1-14 can only access higher privilege levels if an **enable password** has been configured for the level the user tries to access and the user enters that password. A user at privilege level 1 can access the majority of show commands. A user at privilege level 7 can access the majority of show commands including platform show commands. Privilege Level 15 (to access the Privileged Exec command mode) is required to access configuration commands as well as show commands in Privileged Exec.
`password`	A password that the user must enter when logging in.	
	`8`	Specifies that you are entering a password as a string that has already been encrypted, instead of entering a plain-text password. The running-config displays the new password as an encrypted string even if password encryption is turned off. Note that the user enters the plain-text version of the password when logging in.
	<password>	The user’s password. The password can be up to 23 characters in length and include characters from up to four categories. The password categories are: ■ uppercase letters: A to Z ■ lowercase letters: a to z ■ digits: 0 to 9 ■ special symbols: all printable ASCII characters not included in the previous three categories. The question mark ? cannot be used as it is reserved for help functionality.

Mode Global Configuration

Default The privilege level is 1 by default. Note the default is not shown in running-config output.

Usage An intermediate CLI security level (privilege level 7 to privilege level 14) allows a CLI user access to the majority of show commands, including the platform show commands that are available at privilege level 1 to privilege level 6). Note that some show commands, such as show running-configuration and show startup-configuration, are only available at privilege level 15.

A privilege level of 0 can be set for port authentication purposes from a RADIUS server.

Examples To create the user `bob` with a privilege level of `15`, for all show commands including show running-configuration and show startup-configuration and to access configuration commands in Privileged Exec command mode, and the password `bobs_secret`, use the commands:

```
awplus# configure terminal
awplus(config)# username bob privilege 15 password
bobs_secret
```

To create a user `junior_admin` with a privilege level of `7`, for intermediate CLI security level access for most show commands, and the password `show_only`, use the commands:

```
awplus# configure terminal
awplus(config)# username junior_admin privilege 7
password show_only
```

Related Commands
enable password
security-password minimum-categories
security-password minimum-length

Chapter 6: Creating and Managing Files

Introduction 6.2
USB Support 6.2
Working with Files 6.2
- Listing Files 6.2
- Displaying the Contents of Configuration and Text Files 6.4
- Navigating Through the File System 6.4
- Using the Editor 6.6

Creating and Using Configuration Files 6.8
- Creating a Configuration File 6.8
- Specifying the Start-Up Configuration Script 6.8
- Working with Configuration Files 6.9
- The Configuration File Fallback Order 6.10

Copying Files to and from Your Device 6.12
- URL Syntax 6.12
- Copying Files 6.13
- Copying from a Server to Running Configuration 6.17

The Autoboot Feature 6.18
- Restoring a Switch Using Autoboot from External Media 6.19
- Configure Autoboot 6.21

Introduction

This chapter provides information on:

- **USB Support**
- **Working with Files**
- **Creating and Using Configuration Files**
- **Copying Files to and from Your Device**

USB Support

Your switch supports both standard and secure USB storage devices.

USB storage devices used as backup memory can be easily pulled out of a switch. You can use Secure USB storage devices to protect this data in the event that it is mislaid or in unauthorized hands. Secure USB devices provide password (PIN)-protected encryption to the data they store.

Note that if the switch reboots, the Secure USB key will be locked.

After a VCStack failover, when a stack member recovers, the Secure USB device cannot be accessed until it is unlocked.

Working with Files

The AlliedWare Plus™ OS lets you create directory trees for file storage. This section shows:

- **"Listing Files" on page 6.2**—listing files and seeing how much free space you have
- **"Displaying the Contents of Configuration and Text Files" on page 6.4**
- **"Navigating Through the File System" on page 6.4**—identifying the current directory, changing directories, and creating and deleting directories
- **"Using the Editor" on page 6.6**

Flash compaction

The Flash memory on the switch automatically compacts itself to recover space available from deleted files. The switch only does this when necessary, and not every file deletion causes Flash compaction. Flash compaction can occur after a file of any size is added to or deleted from the switch.

Caution While Flash is compacting, the console is unresponsive. Do not restart the switch, as interrupting Flash compaction can damage files.

Listing Files

To list files, enter Privileged Exec mode and enter the command:

```
awplus# dir
```

The output lists files and directories in order of modification date, descending. It looks like this:

```
-rw-      534 Jul 12 2011 17:52:50  stp.cfg
-rw-      534 Jul 12 2011 17:12:50  example.cfg
-rw- 12429011 Jul 12 2011 16:26:06  x510-5.4.4-3.5.rel
```

Listing files including hidden system files

The **dir** command does not list all files—it hides system files and directories because users generally do not need to create or edit them. To list all files including system files, enter Privileged Exec mode and enter the command:

```
awplus# dir all
```

The output looks like this:

```
drwx        0 Jul 12 2011 17:16:32  ./
-rw-      401 Jul 12 2011 17:16:32  example.cfg
-rw-      534 Jul 12 2011 17:52:50  stp.cfg
-rw- 12429011 Jul 12 2011 16:26:06  x510-5.4.4-3.5.rel
drwx      216 Jul  9 2011 11:31:18  ../
drwx        0 Jun 13 2011 04:31:51  .configs/
-rw-       17 Jun 13 2011 04:27:27  .release
drwx        0 Jul 10 2011 23:40:00  .ssh/
```

The hidden files and directories begin with a dot.

Seeing information about the filesystem

To display information about the different memory types on the switch, enter Privileged Exec mode and enter the command:

```
awplus# show file systems
```

The output includes the amount of free memory and the prefix you type to access that memory type, and looks like this:

```
 Size(b)  Free(b)  Type   Flags  Prefixes   S/D/V   Lcl/Ntwk  Avail
-----------------------------------------------------------------
 126.0M   106.4M   flash     rw  flash:     static  local      Y
      -        -   system    rw  system:    virtual local      -
  10.0M     9.8M   debug     rw  debug:     static  local      Y
 499.0K   404.0K   nvs       rw  nvs:       static  local      Y
      -        -   usbstick  rw  usb:       dynamic local      N
      -        -   tftp      rw  tftp:      -       network    -
      -        -   scp       rw  scp:       -       network    -
      -        -   sftp      ro  sftp:      -       network    -
      -        -   http      ro  http:      -       network    -
```

Listing files in a subdirectory

To list the contents of a directory, enter Privileged Exec mode and enter the command:

```
awplus# dir <directory-name>
```

Tip You can specify the directory with or without a / after the directory name.

Example To display the contents of a directory called “example”, enter the command:

```
awplus# dir example
```

Listing files in NVS memory or on a USB storage device

To list the contents of a directory on a USB storage device, enter the command:

```
awplus# dir usb:<directory-name>
```

Example To display the contents of a directory in NVS called “example”, enter the command:

```
awplus# dir nvs:example
```

Displaying the Contents of Configuration and Text Files

To display the contents of a file, enter Privileged Exec mode and enter the command:

```
awplus# show file <filename>
```

Example To display the contents of the file called “example.cfg”, enter the command:

```
awplus# show file example.cfg
```

Navigating Through the File System

Showing the current directory

To see which directory you are currently in, enter Privileged Exec mode and enter the command:

```
awplus# pwd
```

For the top-level directory, the output looks like this:

```
flash:/
```

Changing directories

To change to another directory, enter Privileged Exec mode and enter the command:

```
awplus# cd <directory-name>
```

To go to a directory one level higher in the directory tree, enter the command:

```
awplus# cd ..
```

Example To change to a directory called “example”, enter the command:

```
awplus# cd example
```

To go up one level, which returns you to the top level directory, enter the command:

```
awplus# cd ..
```

Changing to a directory in NVS memory or on a USB storage device

To change to the top-level directory in the NVS memory filesystem, enter Privileged Exec mode and enter the command:

```
awplus# cd nvs:
```

To change to the top-level directory on a USB storage device, enter the command:

```
awplus# cd usb:/
```

Next, you can change to other directories by entering the command:

```
awplus# cd <directory-name>
```

Alternatively, you can go straight from Flash to a subdirectory in the alternative filesystem, by entering one of the commands:

```
awplus# cd nvs:<directory-name>
```

```
awplus# cd usb:/<directory-name>
```

To return to the Flash filesystem, enter the command:

```
awplus# cd flash:/
```

Example To change to the directory within NVS called “example”, enter the command:

```
awplus# cd nvs:example
```

To go up one level, which returns you to the top-level directory of NVS memory, enter the command:

```
awplus# cd ..
```

Creating new directories

To create a directory, enter Privileged Exec mode and enter the command:

```
awplus# mkdir <directory-name>
```

Example To make a directory called “example” within the Flash filesystem, enter the command:

```
awplus# mkdir example
```

Deleting directories

To delete an empty directory, enter Privileged Exec mode and enter the command:

```
awplus# rmdir <directory-name>
```

To delete a directory and all its contents, enter Privileged Exec mode and enter the command:

```
awplus# delete recursive <directory-name>
```

The switch prompts you for confirmation.

Example To delete an empty directory called “example” from within the Flash filesystem, enter the command:

```
awplus# rmdir example
```

Using the Editor

The inbuilt editor is JOE (Joe’s Own Editor).

To edit an existing file, enter Privileged Exec mode and enter the command:

```
awplus# edit <filename>
```

To open the editor with an empty file, enter the command:

```
awplus# edit
```

When you save the new file, you may need to specify the filesystem to store it on. For Flash, use **flash:/<*filename*>**.

Using JOE To format and manipulate text in JOE, you use control-character sequences. The following table summarizes a few useful sequences—for details, see:

joe-editor.sourceforge.net/manpage.html.

Function	Control-character sequence
Access the help	Ctrl-K-H
Save the file without exiting (for new files, this prompts for a filename)	Ctrl-K-D
Save the file and exit (this prompts for a filename)	Ctrl-K-X
Exit without saving the file	Ctrl-C
Go to the beginning of the file	Ctrl-K-U
Go to the end of the file	Ctrl-K-V
Go up one full screen of text in the file	Ctrl-U
Go down one full screen of text in the file	Ctrl-V
Select a block of text:	
Mark the beginning of the block	Ctrl-K-B
Mark the end of the block	Ctrl-K-K
Copy and paste a selected block of text	Place cursor at destination then enter Ctrl-K-C
Move a selected block of text	Place cursor at destination then enter Ctrl-K-M
Delete a selected block of text	Ctrl-K-Y

Creating and Using Configuration Files

This section provides instructions on:

- **Creating a Configuration File**
- **Specifying the Start-Up Configuration Script**
- **Working with Configuration Files**

Creating a Configuration File

A **configuration file** is a text file that contains a sequence of standard commands for a specific purpose. Configuration files have a **.cfg** extension.

You can create and edit configuration files on your device by:

- saving the dynamic configuration on the device, known as the **running-config** (see **"Working with Configuration Files"**). Use the command:

```
awplus# copy running-config (destination-URL)
```

  where URL specifies a file in Flash. If you do not specify a file name, the switch names the file **default.cfg**.

- using the device's text editor. Use the command:

```
awplus# edit (source-URL)
```

  where **source-URL** is the name of the copied file in Flash memory.

- creating a file on a remote PC, then copying it to onto your device. See **"Copying Files"** for more information about using the **copy** commands.

Once you have created a configuration file, you can use it as the **startup-config** file. See **"Specifying the Start-Up Configuration Script"** for more information.

Specifying the Start-Up Configuration Script

When you restart your device, or when it automatically restarts, it executes the pre-configured commands in a configuration script known as the **boot config** or **startup-config** file.

When you first start your device, the script set as the startup-config file is **default.cfg**. If desired, you can overwrite **default.cfg** with another configuration. Alternatively, you can change the startup-config by specifying a new file as the startup-config. Use the command:

```
awplus(config)# boot config-file backup URL
```

where **URL** specifies the name and location of a configuration file. At the next restart, the device executes the commands in the specified file.

You can specify that the configuration file is either in the Flash or USB storage device filesystem. However, if you specify that the configuration file is on a USB storage device then you must first create a backup configuration file stored in Flash.

To specify a backup configuration file, use the command:

```
awplus(config)# boot config-file backup backup URL
```

where **URL** specifies the name and location of a configuration file.

You can change the content of the file set as the startup-config file by:

- entering commands directly into the CLI, then saving this configuration using the command:

  ```
  awplus# copy running-config startup-config
  ```

  This command saves the device's dynamic configuration into the file that is currently configured as the startup-config file.

- writing commands into a configuration file (see **"Creating a Configuration File"** below), then using the command:

  ```
  awplus# copy SOURCE-URL startup-config
  ```

  This command saves the script from the source file into the file that is currently configured as the startup-config file.

To display the name of the configuration file that is set to execute when the device restarts, enter the command:

```
awplus# show boot
```

To see the commands in the startup-config file, use the command:

```
awplus# show startup-config
```

To erase the file set as the startup-config file, use the command:

```
awplus# erase startup-config
```

At the next restart that occurs after you've erased the file, the device loads the configuration in the file **default.cfg**. This file is set on the system as a backup configuration file that loads if no other file is set as the startup-config file.

Working with Configuration Files

When you use the CLI or GUI to configure your device, it stores this dynamic configuration as a list of commands called the **running-config**. To view the device's running-config, use the command:

```
awplus# show running-config
```

If you turn off the device or restart it, any unsaved changes to the running-config are lost. To save the running-config as a configuration script, use the command:

```
awplus# copy running-config destination-url
```

You may have many configuration files. Storing them on a device allows you to keep a backup device with configuration scripts for every device in the network to speed up network recovery time. Multiple scripts also let you test new configuration scripts before setting them as the startup-config. For example, to test a new script named test.cfg, enter the command:

```
awplus# copy flash:/test.cfg running-config
```

This allows you to run a configuration file any time without restarting the device, by replacing the system's current dynamic configuration with the script in the configuration file. However, note that some commands require you to restart the device before they can take effect, such as the **platform** commands.

You can also set a trigger to automatically execute a configuration script when a predetermined event occurs. For information about creating triggers, see **Chapter 101, Triggers Introduction**.

The Configuration File Fallback Order

The configuration fallback order is: configuration file, backup configuration file, default configuration file and then the factory default configuration. It is important to note the there is a distinction in system behavior between when writing to the startup-config file and when the system boots up.

When you copy a configuration script from a source file into the startup-config file the system will write to the first file that is configured. Potentially, this means that if a configuration file and a backup configuration file are not set you will write to the default.cfg.

At system startup the device goes through the fallback sequence until it finds a file that exists. For example, if the configuration file is not found then the backup configuration file becomes the current boot configuration, or startup-config, and so on. In the output displayed by the **show boot** command, the **Current boot config** parameter shows the startup-config file that the switch will load during the next boot cycle. The fallback sequence when configuration files are deleted is shown below in output from the **show boot** command.

In the example output below, the current boot configuration file, **my.cfg**, is set on the USB storage device. This is the startup-config file that the device loads at the next boot cycle.

```
awplus#show boot
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : usb:/x510-5.4.4-3.5.rel
Backup  boot image : flash:/x510-5.4.4-3.5.rel
Default boot config: flash:/default.cfg
Current boot config: usb:/my.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file exists)
```

In the example output below, the **no boot-config** command has been used to delete the configuration file **my.cfg** on the USB storage device. The backup configuration file **backup.cfg** in Flash then becomes the current boot config.

```
awplus#show boot
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : usb:/x510-5.4.4-3.5.rel
Backup  boot image : flash:/x510-5.4.4-3.5.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/backup.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file exists)
```

In the example output below, the **no boot-config backup** command has been used to delete the backup configuration file **backup.cfg**. The default configuration file **default.cfg** then becomes the current boot config.

```
awplus#show boot
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rell
Current boot image : usb:/x510-5.4.4-3.5.rel
Backup  boot image : flash:/x510-5.4.4-3.5.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/default.cfg (file exists)
Backup  boot config: Not set
```

If the current boot configuration file is set on a USB storage device and this device has been removed from the switch, the **Current boot config** parameter field indicates that this file cannot be found, as shown in the following example output.

```
awplus#show boot
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : usb:/x510-5.4.4-3.5.rel
Backup  boot image : flash:/x510-5.4.4-3.5.rel
Default boot config: flash:/default.cfg
Current boot config: usb:/my.cfg (file not found)
Backup  boot config: flash:/backup.cfg (file exists)
```

At system startup the switch will load the backup configuration file as the startup-config.

Copying Files to and from Your Device

This section provides instructions on:

- **URL Syntax**
- **Copying Files**

URL Syntax

Many of the file management commands use the placeholder "URL" to represent the name and location of the file that you want to act on. The following table explains the syntax of this URL for each different type of file location.

When you copy a file...	Use this syntax:	Example:
Copying in local Flash memory	`flash:[/][<directory>/]<filename>`	To specify a file in the configs directory in Flash: `flash:configs/example.cfg`
Copying to or from a USB storage device	`usb:[/][<directory>/]<filename>`	To specify a file in the top-level directory of the USB stick: `usb:example.cfg`
Copying with Hypertext Transfer Protocol (HTTP)	`http://[[<username>:<password>]@] {<hostname>|<host-ip>}[/<filepath>]/<filename>`	To specify a file in the configs directory on the server: `http://www.company.com/configs/example.cfg`
Copying with Trivial File Transfer Protocol (TFTP)	`tftp://[[<location>]/<directory>]/<filename>`	To specify a file in the top-level directory of the server: `tftp://172.1.1.1/example.cfg`
Copying with Secure Copy (SCP)	`scp://<username>@<location>[/<directory>][/<filename>]`	To specify a file in the configs directory on the server, logging on as user "bob": `e.g. scp://bob@10.10.0.12/configs/example.cfg`
Copying with SSH File Transfer Protocol (SFTP)	`sftp://[[<location>]/<directory>]/<filename>`	To specify a file in the top-level directory of the server: `sftp://10.0.0.5/example.cfg`
Copying to or from stack member Flash	`<hostname>-<stack_ID>/flash:[/][<directory>/]<stack_member_filename>`	To specify a file in the configs directory on member 2 of a stack named vcstack: `vcstack-2/flash:/configs/example.cfg`

The URL can include characters from up to four categories. The URL categories are:

1. uppercase letters: A to Z
2. lowercase letters: a to z
3. digits: 0 to 9
4. special symbols: all printable ASCII characters not included in the previous three categories. Including the following characters:

   « -
   « /
   « .
   « _
   « @
   « "
   « '
   « *
   « :
   « ~
   « ?

Do not use spaces or parentheses within filenames. Use hyphens or underlines instead.

Copying Files

To copy files, use the **copy** commands. These commands allow you to copy files:

- between different memory types attached to your device. Use the command:

  ```
  awplus# copy <local-source> <local-destination> <filename>
  ```

  See **"Copying within a filesystem"** and **""** for further details.
- across a serial connection using ZMODEM. Use the command:

  ```
  awplus# copy zmodem
  ```

  See **"Copying with ZMODEM"** for further details.

- from your device onto a remote device, or to your device from a remote device. To copy a file across an interface with IP configured, use the command:

```
awplus# copy SOURCE-URL DESTINATION-URL
```

To copy files across these interfaces you can use the following protocols:

- **"Copying with Hypertext Transfer Protocol (HTTP)"**
- **"Copying with Trivial File Transfer Protocol (TFTP)"**
- **"Copying with Secure Copy (SCP)"**
- **"Copying with SSH File Transfer Protocol (SFTP)"**

Copying within a filesystem

Within a directory

To copy a file within the same directory, enter Privileged Exec mode and enter the command:

```
awplus# copy <source-filename> <destination-filename>
```

If the file already exists, the switch asks whether to overwrite it, with a message like this:

```
Overwrite flash:/example.cfg? (y/n)[n]:
```

To overwrite, press the "y" key then the Enter key.

Between directories

To copy a file to another directory within the same filesystem, enter the command:

```
awplus# copy <source-filename> <directory-name>
```

The / after the directory name is required. Otherwise the switch displays an error ("37: Destination file is a directory").

The switch then prompts you for the destination filename. To give the copy a new name, type the name at the prompt. You can include directory names in the path.

To use the same filename as the original, press the Enter key (do not press the "y" key—that names the copy "y").

Example

To put a copy of example.cfg into the example directory, enter the command:

```
awplus# copy example.cfg example/
```

The prompt and messages look like this:

```
Enter destination file name [example.cfg]:
Copying from source file, please wait...
Copying to destination file, please wait...
0: Successful operation
```

Copying to and from NVS or a USB storage device

In a stacked environment you can only access `flash` and `nvs` using the stack member file path. To access a USB storage device on a backup stack member, use the **remote-login** command on page 111.7.

To copy between filesystems, you need to specify the filesystem prefix (`nvs:` or `usb:`).

For example, to copy from Flash to NVS when your current directory is the top-level Flash directory, enter Privileged Exec mode and enter the command:

```
awplus# copy <source-filename> nvs:sd:
```

For example, to copy from Flash to the USB storage device when your current directory is the top-level Flash directory, enter Privileged Exec mode and enter the command:

```
awplus# copy <source-filename> usb:
```

The switch prompts you for the filename, as described in the previous section.

To copy from NVS to Flash when your current directory is the top-level Flash directory, enter the command:

```
awplus# copy nvs:<source-filename> <destination-filename>
```

Example To copy the file “example.txt” from the directory called “example” in NVS, to the top level of Flash, enter the command:

```
awplus# copy nvs:example/example.txt example.txt
```

Copying with ZMODEM

ZMODEM allows you to copy files from a network host over an asynchronous port. Use the command:

```
awplus# copy zmodem
```

to open Minicom and transfer a file. Alternatively you can specify the file name within the command:

```
awplus# copy SOURCE-URL zmodem
```

For example, to copy the file “july.cfg” from Flash memory using ZMODEM, use the command:

```
awplus# copy flash:/july.cfg zmodem
```

Copying with Hypertext Transfer Protocol (HTTP)

You device has a built-in HTTP client. The HTTP client enables the device to act as a browser by sending HTTP “get” or “post” requests to an HTTP server. The client is enabled by default.

For example, to load the file “bob.key” onto Flash from the security directory on the web server at www.company.com, use the command:

```
awplus# copy http://www.company.com/security/bob.key
       flash:/bob.key
```

Copying with Trivial File Transfer Protocol (TFTP)

TFTP runs over User Datagram Protocol (UDP). It is simpler and faster than FTP but has minimal capability, such as no provisions for user authentication.

To copy a file from a TFTP server to Flash memory, enter Privileged Exec mode and enter the command:

```
awplus# copy tftp flash
```

Note

You can specify the server and filename in the command instead of waiting for prompts. Use a format like the following:

```
copy tftp://172.1.1.1/example.cfg flash
```

The switch prompts you for the:

- TFTP server hostname (you can enter its IP address instead)
- source filename on the TFTP server
- destination filename in Flash on the switch

To copy a file from Flash to a TFTP server, enter the command:

```
awplus# copy flash tftp
```

Follow the prompts for source filename, server, and destination filename.

If the file is not in the top level of the TFTP server, include the path as part of the filename.

Example To copy example.cfg to the TFTP server at 172.1.1.1, enter the command:

```
awplus# copy flash tftp
```

The prompts, responses, and messages look like this:

```
Enter source file name []:example.cfg
Enter destination host name []:172.1.1.1
Enter destination file name [example.cfg]:
Copying from source file, please wait...
Copying to destination file, please wait...
0: Successful operation
```

To load the file “bob.key” from a TFTP server, where the file is in the folder “security”, use the command:

```
awplus# copy tftp://security/bob.key flash:/bob.key
```

Copying with Secure Copy (SCP)

Secure Copy (SCP) provides a secure way to copy files to and from a remote device using SSH. The AlliedWare Plus™ OS includes both a SSH server and a SSH client. You must enable the SSH server before your device accepts connections from SCP clients. See the **Chapter 76, Secure Shell (SSH) Introduction** for more information.

For example, to load the file “beth.key” onto Flash from the key directory on a remote SSH server at 10.10.0.12, using the username “bob”, use the command:

```
awplus# copy scp://bob@10.10.0.12/key/beth.key
       flash:/beth.key
```

Copying with SSH File Transfer Protocol (SFTP)

SSH File Transfer Protocol (SFTP) provides a secure way to copy files onto your device from a remote device. The AlliedWare Plus™ OS includes both a SSH server and a SSH client. SFTP provides additional features from SCP, such as allowing you to manipulate the remote files, and halt or resume file transfers without closing the session.

For example, to load the file “rei.cfg” onto Flash memory from the remote server at 10.0.0.5, use the command:

```
awplus# copy sftp://10.0.0.5/rei.cfg flash:/rei.cfg
```

Copying from a Server to Running Configuration

Use the **copy tftp** variant of the **copy running-config** command on page 7.14 to load a configuration file from a server to the running configuration of the switch.

The configuration will be added to the running configuration as if the commands were typed in the command line interface.

The resulting configuration file will be a combination of the previous running configuration and the loaded configuration file. The loaded configuration file has precedence.

The Autoboot Feature

The Autoboot feature enables your switch to automatically load a specific release file and/or configuration file from external media, such as USB storage device, into Flash memory, providing there is enough free space available.

If there is not enough free space, the Autoboot feature will exit and booting will revert to what was previously set by the CLI. This feature is enabled only the first time the device is powered up in the field. Subsequently, the Autoboot feature is disabled by default.

The Autoboot feature minimizes network downtime by avoiding the need for manual configuration of a replacement device.

If you use prepared external media for the first time boot, the Autoboot feature gives you the ability to easily ensure the device boots with your desired release and configuration files. You must prepare the external media for this purpose using an initiation file, `autoboot.txt`, and accompanying release and configuration files.

Use the **create autoboot** command to create an `autoboot.txt` file on external media. This command will automatically ensure that the keys and values that are expected in this file are correct. After the file is created the command will copy the current release and configuration files across to the external media. The `autoboot.txt` file is read/writable by any desktop operating system currently supported by the AlliedWare Plus™ Operating System. Note that the external media file system is not case sensitive.

When the Autoboot feature is enabled, the device on boot-up:

- checks for a special file called `autoboot.txt` on external media, and if this file exists,
- checks in the file for the "key=value" pair "`Copy_from_external_media_enabled=yes`", and if this enable flag is set,
- loads the release file and/or configuration file from external media.

An example of a valid `autoboot.txt` file is shown in **Figure 6-1** below.

Figure 6-1: Example autoboot.txt file

```
; J Smith,  x510-28GTX,  14 Aug 2012
[AlliedWare Plus]
Copy_from_external_media_enabled=yes
Boot_Release=x510-5.4.4-3.5.rel
Boot_Config=network1.cfg
```

We recommend that no directories are present on external media used to hold the `autoboot.txt` file. In addition, large numbers of files on external media may slow the booting process.

Note The Autoboot feature is not supported in a stacked configuration.

Do not remove external media part way through the copy process as this may leave the device in an unstable state.

Configuration files placed on external media reduce security. Therefore, ensure adequate security precautions are taken with external media holding configuration files.

Configuration commands that rely on the presence of a feature license will fail when executed in the replacement switch if the replacement switch does not have the same feature license present.

Restoring a Switch Using Autoboot from External Media

The example below describes the sequence of events when a switch in the field fails and is restored using this feature:

1. Using the **create autoboot** command, a network engineer has previously manually created a restore external media device, such as a USB storage device. The external media device contains the following components:
   - An `autoboot.txt` file with required contents
   - An appropriate release file
   - A configuration file
2. A switch fails in the field.
3. A replacement switch of same model is installed.
4. The previously created external media device is placed into the replacement switch.
5. The switch powers up using its pre-installed release if present. It automatically checks the external media device for the `autoboot.txt` file.
6. The switch finds a valid `autoboot.txt` file on the external media device, with the value “Copy_from_external_media_enabled” set. The release file and configuration file both exist on the external media device.
7. The MD5sum of pre-installed Flash release file is compared to the MD5sum of the release file stored in the external media device. If they do not match, because the release file in the replacement switch is either missing or different, then the release is restored from the external media device. If the release files already match, then the release file is not copied from the external media device.
8. The MD5sum of the Flash configuration file `default.cfg` (if pre-installed in the replacement switch) is compared to the MD5sum of the configuration file stored in the external media device. If they do not match, because the configuration file in the replacement switch is either missing or different, then the configuration file is

restored from the external media device. If the configuration files already match, then the configuration file is not copied from the external media device.

9. The memory space available in the switch Flash is checked to ensure the release and configuration files stored in the external media device will fit. If there is not enough space the Autoboot feature will exit.

10. The release file and configuration files are automatically copied from the external media device to switch Flash memory. The switch release and configuration files are updated to contain the appropriate names.

11. The switch is automatically rebooted.

12. The replacement switch is now running the restored release and configuration files. Subsequent reboots are based on the restored release and configuration files stored in the switch Flash memory.

13. If you want to Autoboot from external media on this specific switch in the future, you must now manually enable the Autoboot feature in the configuration menu via the **autoboot enable** command. This command resets the enable flag stored internally in the switch NVS memory.

Configure Autoboot

This section describes the commands used to configure the Autoboot feature.

Table 6-1: Configuration procedures for the Autoboot feature

Create an Autoboot file (autoboot.txt)	
`awplus#`	
`create autoboot [usb]`	Create an **autoboot.txt** file on external media.
Enable the Autoboot feature	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`autoboot enable`	The Autoboot feature is enabled by default the first time the device is powered up in the field. Use this command to enable the feature subsequently.
Disable the Autoboot feature	
`awplus(config)#`	
`no autoboot enable`	Use this command to disable the Autoboot feature.
Display Autoboot configuration and status	
`awplus#`	
`show autoboot`	Display detailed information about the current Autoboot configuration and status.
`awplus#`	
`show boot`	Display the status of the Autoboot feature; either enabled or disabled.

Chapter 7: File Management Commands

Introduction 7.3
URL Syntax and Keyword Usage 7.3
Command List 7.5
autoboot enable 7.5
boot config-file 7.6
boot config-file backup 7.8
boot system 7.9
boot system backup 7.11
cd 7.12
copy current-software 7.12
copy debug 7.13
copy running-config 7.14
copy startup-config 7.15
copy (URL) 7.16
copy zmodem 7.18
create autoboot 7.19
delete 7.20
delete debug 7.21
dir 7.22
edit 7.24
edit (URL) 7.25
erase startup-config 7.26
mkdir 7.27
move 7.28
move debug 7.29
pwd 7.30
rmdir 7.31
show autoboot 7.32
show boot 7.33
show file 7.35
show file systems 7.36
show running-config 7.37
show running-config access-list 7.39
show running-config as-path access-list 7.40
show running-config dhcp 7.41
show running-config full 7.42
show running-config interface 7.43
show running-config ip pim dense-mode 7.46
show running-config ip pim sparse-mode 7.47
show running-config ip route 7.48
show running-config ipv6 access-list 7.49
show running-config ipv6 mroute 7.50
show running-config ipv6 prefix-list 7.51
show running-config ipv6 route 7.52
show running-config key chain 7.53
show running-config lldp 7.54
show running-config power-inline 7.55

show running-config prefix-list 7.56
show running-config route-map 7.57
show running-config router 7.58
show running-config router-id 7.59
show running-config security-password 7.60
show startup-config 7.61
show version 7.62
write file 7.63
write memory 7.63
write terminal 7.63

Introduction

This chapter provides an alphabetical reference of AlliedWare Plus™ OS file management commands.

URL Syntax and Keyword Usage

Many of the commands in this chapter use the placeholder "URL" to represent the name and location of the file that you want to act on. The following table explains the syntax of this URL for each different type of file location.

When you copy a file...	Use this syntax:	Example:
Copying in local Flash memory	`flash:[/][<directory>/]<filename>`	To specify a file in the configs directory in Flash: `flash:configs/example.cfg`
Copying to or from a USB storage device	`usb:[/][<directory>/]<filename>`	To specify a file in the top-level directory of the USB stick: `usb:example.cfg`
Copying with Hypertext Transfer Protocol (HTTP)	`http://[[<username>:<password>]@] {<hostname>|<host-ip>}[/<filepath>]/<filename>`	To specify a file in the configs directory on the server: `http://www.company.com/configs/example.cfg`
Copying with Trivial File Transfer Protocol (TFTP)	`tftp://[[<location>]/<directory>]/<filename>`	To specify a file in the top-level directory of the server: `tftp://172.1.1.1/example.cfg`
Copying with Secure Copy (SCP)	`scp://<username>@<location>[/<directory>][/<filename>]`	To specify a file in the configs directory on the server, logging on as user "bob": `e.g. scp://bob@10.10.0.12/configs/example.cfg`
Copying with SSH File Transfer Protocol (SFTP)	`sftp://[[<location>]/<directory>]/<filename>`	To specify a file in the top-level directory of the server: `sftp://10.0.0.5/example.cfg`
Copying to or from stack member Flash	`<hostname>-<stack_ID>/flash:[/][<directory>/]<stack_member_filename>`	To specify a file in the configs directory on member 2 of a stack named vcstack: `vcstack-2/flash:/configs/example.cfg`

Valid characters

The URL can include characters from up to four categories. The URL categories are:

1. uppercase letters: A to Z
2. lowercase letters: a to z
3. digits: 0 to 9
4. special symbols: all printable ASCII characters not included in the previous three categories. Including the following characters:
   - « -
   - « /
   - « .
   - « _
   - « @
   - « "
   - « '
   - « *
   - « :
   - « ~
   - « ?

Do not use spaces or parentheses within filenames. Use hyphens or underscores instead.

Syntax for directory listings

A leading slash (/) indicates the root of the current filesystem location.

In commands where you need to specify the local filesystem's Flash base directory, you may use **flash** or **flash:** or **flash:/**. For example, these commands are all the same:

- `dir flash`
- `dir flash:`
- `dir flash:/`

Similarly, you can specify the USB storage device base directory with **usb** or **usb:** or **usb:/**

You cannot name a directory or subdirectory **flash**, **nvs**, **usb**, **card**, **tftp**, **scp**, **sftp** or **http.** These keywords are reserved for tab completion when using various file commands.

In a stacked environment you can only access `flash` and `nvs` using the stack member filepath (e.g. `dir awplus-2/flash:/`). To access a USB storage device on a backup stack member, use the **remote-login** command on page 111.7.

Command List

autoboot enable

This command enables the device to restore a release file and/or a configuration file from external media, such as a USB storage device.

When the Autoboot feature is enabled, the device looks for a special file called `autoboot.txt` on the external media. If this file exists, the device will check the key and values in the file and recover the device with a new release file and/or configuration file from the external media. An example of a valid `autoboot.txt` file is shown in Figure 7-1 below.

Figure 7-1: Example autoboot.txt file

```
[AlliedWare Plus]
Copy_from_external_media_enabled=yes
Boot_Release=x510-5.4.4-3.5.rel
Boot_Config=network1.cfg
```

Use the **no** variant of this command to disable the Autoboot feature.

Note This command is not supported in a stacked configuration.

Syntax

```
autoboot enable
no autoboot enable
```

Default The Autoboot feature operates the first time the device is powered up in the field, after which the feature is disabled by default.

Mode Global Configuration

Example To enable the Autoboot feature, use the command:

```
awplus# configure terminal
awplus(config)# autoboot enable
```

To disable the Autoboot feature, use the command:

```
awplus# configure terminal
awplus(config)# no autoboot enable
```

Related Commands

create autoboot
show autoboot
show boot

boot config-file

Use this command to set the configuration file to use during the next boot cycle.

Use the **no** variant of this command to remove the configuration file.

Syntax

```
boot config-file <filepath-filename>
no boot config-file
```

Parameter	Description
<filepath-filename>	Filepath and name of a configuration file. The specified configuration file must exist in the specified filesystem. Valid configuration files must have a **.cfg** extension.

Mode Global Configuration

Usage You can only specify that the configuration file is on a USB storage device if there is a backup configuration file already specified in Flash. If you attempt to set the configuration file on a USB storage device and a backup configuration file is not specified in Flash, the following error message is displayed:

```
% Backup configuration files must be stored in the flash
filesystem
```

For an explanation of the configuration fallback order, see **"The Configuration File Fallback Order" on page 6.10**.

Examples To run the configuration file `branch.cfg` stored on the switch's Flash filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# boot config-file flash:/branch.cfg
```

To remove the configuration file `branch.cfg` stored on the switch's Flash filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# no boot config-file flash:/branch.cfg
```

To run the configuration file `branch.cfg` stored on the switch's USB storage device filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# boot config-file usb:/branch.cfg
```

To remove the configuration file `branch.cfg` stored on the switch's USB storage device filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# no boot config-file usb:/branch.cfg
```

Related Commands

boot config-file backup
boot system
boot system backup
show boot

boot config-file backup

Use this command to set a backup configuration file to use if the main configuration file cannot be accessed.

Use the **no** variant of this command to remove the backup configuration file.

Syntax

```
boot config-file backup <filepath-filename>
no boot config-file backup
```

Parameter	Description
<filepath-filename>	Filepath and name of a backup configuration file. Backup configuration files must be in the Flash filesystem. Valid backup configuration files must have a **.cfg** extension.
`backup`	The specified file is a backup configuration file.

Mode Global Configuration

Usage For an explanation of the configuration fallback order, see **"The Configuration File Fallback Order" on page 6.10**.

Examples To set the configuration file `backup.cfg` as the backup to the main configuration file, use the commands:

```
awplus# configure terminal
awplus(config)# boot config-file backup flash:/backup.cfg
```

To remove the configuration file `backup.cfg` as the backup to the main configuration file, use the commands:

```
awplus# configure terminal
awplus(config)# no boot config-file backup flash:/backup.cfg
```

Related Commands

boot config-file
boot system
boot system backup
show boot

boot system

Use this command to set the release file to load during the next boot cycle.

Use the **no** variant of this command to remove the release file as the boot file.

Syntax

```
boot system <filepath-filename>
no boot system
```

Parameter	Description
<filepath-filename>	Filepath and name of a release file. The specified release file must exist and must be stored in the root directory of the specified filesystem. Valid release files must have a **.rel** extension.

Mode Global Configuration

Usage You can only specify that the release file is on a USB storage device if there is a backup release file already specified in Flash. If you attempt to set the release file on a USB storage device and a backup release file is not specified in Flash, the following error message is displayed:

```
% A backup boot image must be set before setting a current boot
image on USB storage device
```

In a VCStack configuration, a release file on a USB storage device is accepted only if a card is inserted in all stack members and all stack members have a bootloader version that supports booting from card. If a stack member has a card removed an error message is displayed. For example, if stack member 2 does not have a card inserted the following message is displayed:

```
% Stack member 2 has no card inserted
```

Examples To run the release file `x510-5.4.4-3.5.rel` stored on the switch's Flash filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# boot system flash:/x510-5.4.4-3.5.rel
```

To remove the release file `x510-5.4.4-3.5.rel` stored on the switch's Flash filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# no boot system flash:/x510-5.4.4-3.5.rel
```

To run the release file `x510-5.4.4-3.5.rel` stored on the switch's USB storage device

filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# boot system usb:/x510-5.4.4-3.5.rel
```

To remove the release file `x510-5.4.4-3.5.rel` stored on the switch's USB storage device filesystem the next time the device boots up, use the commands:

```
awplus# configure terminal
awplus(config)# boot system usb:/x510-5.4.4-3.5.rel
```

In a VCStack configuration, if there is not enough space to synchronize the new release across the stack, the boot system command has an interactive mode that prompts you to delete old releases.

```
awplus# configure terminal
awplus(config)# boot system x510-5.4.4-3.5.rel
```

```
Insufficient flash available on stack member-2 (11370496)
 to synchronize file x510-5.4.4-3.5.rel (14821895).

 List of release files on stack member-2
        x510-5.4.4-3.5.rel (14822400)

 Select files to free up space,
 Delete awplus-2/flash:/x510-5.4.4-3.5.rel? (y/n)[n]:y
```

```
awplus(config)# y
```

```
 Deleting selected files, please
wait.................................
 Successful operation
 VCS synchronizing file across the stack, please
wait.....................................
 File synchronization with stack member-2 successfully completed
 [DONE]
```

Related Commands

boot config-file
boot config-file backup
boot system backup
show boot

boot system backup

Use this command to set a backup release file to load if the main release file cannot be loaded.

Use the **no** variant of this command to remove the backup release file as the backup boot file.

Syntax

```
boot system backup <filepath-filename>
no boot system backup
```

Parameter	Description
<filepath-filename>	Filepath and name of a backup release file. Backup release files must be in the Flash filesystem. Valid release files must have a **.rel** extension.
backup	The specified file is a backup release file.

Mode Global Configuration

Examples To specify the file `x510-5.4.4-3.5.rel` as the backup to the main release file, use the commands:

```
awplus# configure terminal
awplus(config)# boot system backup flash:/x510-5.4.4-3.5.rel
```

To remove the file `x510-5.4.4-3.5.rel` as the backup to the main release file, use the commands:

```
awplus# configure terminal
awplus(config)# no boot system backup flash:/x510-5.4.4-
3.5.rel
```

Related Commands

boot config-file
boot config-file backup
boot system
show boot

cd

This command changes the current working directory.

Syntax `cd <directory-url>`

Parameter	Description
<directory-url>	URL of the directory.

Mode Privileged Exec

Example To change to the directory called `images`, use the command:

```
awplus# cd images
```

Related Commands **dir**
pwd
show file systems

copy current-software

This command copies the AlliedWare Plus™ OS software that the device has booted from, to a destination file. Specify whether the destination is Flash or USB when saving the software to the local filesystem.

Syntax `copy current-software <destination-url>`

Parameter	Description
<destination-url>	The URL where you would like the current running-release saved. This command creates a file if no file exists with the specified filename. If a file already exists, then the CLI prompts you before overwriting the file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Example To copy the current software as installed in the working directory with the file name `my-release.rel`, use the command:

```
awplus# copy current-software my-release.rel
```

Related Commands **boot system backup**
show boot

copy debug

This command copies a specified debug file to a destination file. Specify whether the destination is Flash or USB when saving the software to the local filesystem.

Syntax

```
copy debug {<destination-url>|debug|flash|nvs|scp|tftp|usb} {<source-
    url>|debug|flash|nvs|scp|tftp|usb}
```

Parameter	Description
<destination-url>	The URL where you would like the debug output saved. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.
<source-url>	The URL where the debug output originates. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Example To copy debug output to a USB storage device with a filename `my-debug`, use the following command:

```
awplus# copy debug usb:mydebug
```

Output **Figure 7-2: CLI prompt after entering the copy debug command**

```
Enter source file name []:
```

Related Commands

delete debug
move debug

copy running-config

This command copies the running-config to a destination file, or copies a source file into the running-config. Commands entered in the running-config do not survive a device reboot unless they are saved in a configuration file.

Syntax

```
copy <source-url> running-config
copy running-config [<destination-url>]
copy running-config startup-config
```

Parameter	Description
<source-url>	The URL of a configuration file. This must be a valid configuration file with a **.cfg** filename extension. Specify this when you want the script in the file to become the new running-config. The URL can contain the following protocols or location words. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.
<destination-url>	The URL where you would like the current running-config saved. This command creates a file if no file exists with the specified filename. If a file already exists, then the CLI prompts you before overwriting the file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax. If you do not specify a file name, the switch saves the running-config to a file called default.cfg.
`startup-config`	Copies the running-config into the file set as the current startup-config file.

Mode Privileged Exec

Examples To copy the `running-config` into the startup-config, use the command:

```
awplus# copy running-config startup-config
```

To copy the file `layer3.cfg` into the `running-config`, use the command:

```
awplus# copy layer3.cfg running-config
```

To use SCP to copy the running-config as `current.cfg` to the remote server listening on TCP port 2000, use the command:

```
awplus# copy running-config scp://user@server:2000/
       config_files/current.cfg
```

Related Commands

copy startup-config
write file
write memory

copy startup-config

This command copies the startup-config script into a destination file, or alternatively copies a configuration script from a source file into the startup-config file. Specify whether the destination is Flash or USB when loading from the local filesystem.

Syntax

```
copy <source-url> startup-config
```

```
copy startup-config <destination-url>
```

Parameter	Description
<source-url>	The URL of a configuration file. This must be a valid configuration file with a **.cfg** filename extension. Specify this to copy the script in the file into the *startup-config* file. Note that this does not make the copied file the new startup file, so any further changes made in the configuration file are not added to the startup-config file unless you reuse this command. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.
<destination-url>	The destination and filename that you are saving the startup-config as. This command creates a file if no file exists with the specified filename. If a file already exists, then the CLI prompts you before overwriting the file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Examples To copy the file `Layer3.cfg` to the `startup-config`, use the command:

```
awplus# copy Layer3.cfg startup-config
```

To copy the `startup-config` as the file `oldconfig.cfg` in the current directory, use the command:

```
awplus# copy startup-config oldconfig.cfg
```

Related Commands **copy running-config**

copy (URL)

This command copies a file. This allows you to:

- copy files from your device to a remote device
- copy files from a remote device to your device
- copy files stored on Flash memory to or from a different memory type, such as a USB storage device
- create two copies of the same file on your device

Syntax `copy <source-url> <destination-url>`

Parameter	Description
<source-url>	The URL of the source file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.
<destination-url>	The URL for the destination file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Usage The URL can include characters from up to four categories. The URL categories are:

1. uppercase letters: A to Z
2. lowercase letters: a to z
3. digits: 0 to 9
4. special symbols: all printable ASCII characters not included in the previous three categories. Including the following characters:
   - « -
   - « /
   - « .
   - « _
   - « @
   - « "
   - « '
   - « *
   - « :
   - « ~
   - « ?

Do not use spaces or parentheses within filenames. Use hyphens or underlines instead.

Examples

To use TFTP to copy the file `bob.key` into the current directory from the remote server at `10.0.0.1`, use the command:

```
awplus# copy tftp://10.0.0.1/bob.key bob.key
```

To use SFTP to copy the file `new.cfg` into the current directory from a remote server at `10.0.1.2`, use the command:

```
awplus# copy sftp://10.0.1.2/new.cfg bob.key
```

To use SCP with the username `beth` to copy the file `old.cfg` into the directory `config_files` on a remote server that is listening on TCP port 2000, use the command:

```
awplus# copy scp://beth@serv:2000/config_files/old.cfg old.cfg
```

To copy the file `newconfig.cfg` onto your device's Flash from a USB storage device, use the command:

```
awplus# copy usb:/newconfig.cfg flash:/newconfig.cfg
```

To copy the file `newconfig.cfg` to a USB storage device from your device's Flash, use the command:

```
awplus# copy flash:/newconfig.cfg usb:/newconfig.cfg
```

To copy the file `config.cfg` into the current directory from a USB storage device, and rename it to `configtest.cfg`, use the command:

```
awplus# copy usb:/config.cfg configtest.cfg
```

To copy the file `config.cfg` into the current directory from a remote file server, and rename it to `configtest.cfg`, use the command:

```
awplus# copy fserver:/config.cfg configtest.cfg
```

To copy the file `test.txt` from the top level of Flash on stack member 2 to the current directory in the stack master, use the command:

```
awplus# copy awplus-2/flash:/test.txt test.txt
```

Note that you must specify either the NVS or Flash filesystem on the (backup) stack member (`flash:` in this example).

Related Commands

copy zmodem
edit (URL)
show file systems

copy zmodem

This command allows you to copy files using ZMODEM using Minicom. ZMODEM works over a serial connection and does not need any interfaces configured to do a file transfer.

Syntax

```
copy <source-url> zmodem
```

```
copy zmodem
```

Parameter	Description
<source-url>	The URL of the source file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Example To copy the local file `asuka.key` using ZMODEM, use the command:

```
awplus# copy asuka.key zmodem
```

Related Commands

copy (URL)
show file systems

create autoboot

Use this command to create an `autoboot.txt` file on external media. This command will automatically ensure that the keys and values that are expected in this file are correct. After the file is created the **create autoboot** command will copy the current release and configuration files across to the external media. The external media is then available to restore a release file and/or a configuration file to the device.

Syntax `create autoboot [usb]`

Mode Privileged Exec

Example To create an `autoboot.txt` file on external media, use the command:

```
awplus# create autoboot usb
```

Related Commands
autoboot enable
show autoboot
show boot

delete

This command deletes files or directories.

Syntax `delete [force] [recursive] <url>`

Parameter	Description
`force`	Ignore nonexistent filenames and never prompt before deletion.
`recursive`	Remove the contents of directories recursively.
<url>	URL of the file to delete. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Examples To delete the file `temp.cfg` from the current directory, use the command:

```
awplus# delete temp.cfg
```

To delete the read-only file `one.cfg` from the current directory, use the command:

```
awplus# delete force one.cfg
```

To delete the directory `old_configs`, which is not empty, use the command:

```
awplus# delete recursive old_configs
```

To delete the directory `new_configs`, which is not empty, without prompting if any read-only files are being deleted, use the command:

```
awplus# delete force recursive new_configs
```

Related Commands **erase startup-config**
rmdir

delete debug

Use this command to delete a specified debug output file.

Syntax `delete debug <source-url>`

Parameter	Description
<source-url>	The URL where the debug output originates. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Example To delete debug output, use the following command:

```
awplus# delete debug
```

Output **Figure 7-3: CLI prompt after entering the delete debug command**

```
Enter source file name []:
```

Related Commands **copy debug**
move debug

dir

This command lists the files on a filesystem. If no directory or file is specified then this command lists the files in the current working directory.

Syntax `dir [all] [recursive] [sort [reverse] [name|size|time]] [<url>|debug|flash|nvs|usb]`

Parameter	Description
`all`	List all files.
`recursive`	List the contents of directories recursively.
`sort`	Sort directory listing.
`reverse`	Sort using reverse order.
`name`	Sort by name.
`size`	Sort by size.
`time`	Sort by modification time (default).
<url>	URL of the directory or file. If no directory or file is specified, then this command lists the files in the current working directory.
`debug`	Debug root directory
`flash`	Flash memory root directory
`nvs`	NVS memory root directory
`usb`	USB storage device root directory

Mode Privileged Exec

Usage In a stacked environment you can use the CLI on a stack master to access filesystems that are located on another stack member. Refer to the **URL Syntax and Keyword Usage**.

Examples To list the files in the current working directory, use the command:

```
awplus# dir
```

To list the non-hidden files in the root of the Flash filesystem, use the command:

```
awplus# dir flash
```

To list all the files in the root of the Flash filesystem, use the command:

```
awplus# dir all flash:
```

To list recursively the files in the Flash filesystem, use the command:

```
awplus# dir recursive flash:
```

To list the files in alphabetical order, use the command:

```
awplus# dir sort name
```

To list the files by size, smallest to largest, use the command:

```
awplus# dir sort reverse size
```

To sort the files by modification time, oldest to newest, use the command:

```
awplus# dir sort reverse time
```

To list the files within the Flash filesystem for stack member 2, use the command:

```
awplus# dir awplus-2/flash:/
```

Note that you must specify the filesystem on the stack member (`flash` in this example).

Related Commands **cd**
pwd

edit

This command opens a text file in the AlliedWare Plus™ text editor. Once opened you can use the editor to alter to the file.

If a filename is specified and it already exists, then the editor opens it in the text editor.

If no filename is specified, the editor prompts you for one when you exit it.

Before starting the editor make sure your terminal, terminal emulation program, or Telnet client is 100% compatible with a VT100 terminal. The editor uses VT100 control sequences to display text on the terminal.

For more information about using the editor, including control sequences, see **"Using the Editor" on page 6.6**.

Syntax `edit [<filename>]`

Parameter	Description
<filename>	Name of a file in the local Flash filesystem.

Mode Privileged Exec

Examples To create and edit a new text file, use the command:

```
awplus# edit
```

To edit the existing configuration file `myconfig.cfg` stored on your device's Flash memory, use the command:

```
awplus# edit myconfig.cfg
```

Related Commands **edit (URL)**
show file

edit (URL)

This command opens a remote text file as read-only in the AlliedWare Plus™ text editor.

Before starting the editor make sure your terminal, terminal emulation program, or Telnet client is 100% compatible with a VT100 terminal. The editor uses VT100 control sequences to display text on the terminal.

Syntax `edit <url>`

Parameter	Description
<url>	The URL of the remote file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Usage The URL can include characters from up to four categories. The URL categories are:

1. uppercase letters: A to Z
2. lowercase letters: a to z
3. digits: 0 to 9
4. special symbols: all printable ASCII characters not included in the previous three categories. Including the following characters:
   - « -
   - « /
   - « .
   - « _
   - « @
   - « "
   - « '
   - « *
   - « :
   - « ~
   - « ?

Do not use spaces or parentheses within filenames. Use hyphens or underlines instead.

Example To view the file `bob.key` stored in the security directory of a TFTP server, use the command:

```
awplus# edit tftp://security/bob.key
```

Related Commands
copy (URL)
edit
show file

erase startup-config

This command deletes the file that is set as the startup-config file, which is the configuration file that the system runs when it boots up.

At the next restart, the device loads the default configuration file, default.cfg. If default.cfg no longer exists, then the device loads with the factory default configuration. This provides a mechanism for you to return the device to the factory default settings.

Syntax

```
erase startup-config
```

Mode

Privileged Exec

Example

To delete the file currently set as the startup-config, use the command:

```
awplus# erase startup-config
```

Related Commands

boot config-file backup
copy running-config
copy startup-config
show boot

mkdir

This command makes a new directory.

Syntax `mkdir <url>`

Parameter	Description
<url>	URL of the directory that you are creating.

Mode Privileged Exec

Usage You cannot name a directory or subdirectory **flash**, **nvs**, **usb**, **card**, **tftp**, **scp**, **sftp** or **http.** These keywords are reserved for tab completion when using various file commands.

Example To make a new directory called `images` in the current directory, use the command:

```
awplus# mkdir images
```

Related Commands **cd**
dir
pwd

move

This command renames or moves a file.

Syntax `move <source-url> <destination-url>`

Parameter	Description
<source-url>	The URL of the source file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.
<destination-url>	The URL of the destination file. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Examples To rename the file `temp.cfg` to `startup.cfg`, use the command:

```
awplus# move temp.cfg startup.cfg
```

To move the file `temp.cfg` from the root of the Flash filesystem to the directory `myconfigs`, use the command:

```
awplus# move temp.cfg myconfigs/temp.cfg
```

Related Commands
delete
edit
show file
show file systems

move debug

This command moves a specified debug file to a destination debug file.

Specify whether the destination is Flash or USB when saving the software to the local filesystem.

Syntax

```
move debug {<destination-url>|debug|flash|nvs|usb} {<source-url>|
    debug|flash|nvs|usb}
```

Parameter	Description
<destination-url>	The URL where you would like the debug output moved to. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.
<source-url>	The URL where the debug output originates. See **"URL Syntax and Keyword Usage" on page 7.3** for valid URL syntax.

Mode Privileged Exec

Example To move debug output onto a USB storage device with a filename `my-debug`, use the following command:

```
awplus# move debug usb:my-debug
```

Output **Figure 7-4: CLI prompt after entering the move debug command**

```
Enter source file name []:
```

Related Commands

copy debug
delete debug

pwd

This command prints the current working directory.

Syntax `pwd`

Mode Privileged Exec

Example To print the current working directory, use the command:

```
awplus# pwd
```

Related Commands cd

rmdir

This command removes a directory. The directory must be empty for the command to work unless the optional **force** keyword is used to remove all subdirectories or files in a directory.

Syntax `rmdir [force] <url>`

Parameter	Description
`force`	Optional keyword that allows you to delete any directories that are not empty and may contain files or subdirectories.
`<url>`	The URL of the directory.

Mode Privileged Exec

Usage In a stacked environment you can use the CLI on a stack master to access filesystems that are located on another stack member. Refer to the **URL Syntax and Keyword Usage**.

Examples To remove the directory `images` from the top level of the Flash filesystem, use the command:

```
awplus# rmdir flash:/images
```

To force the removal of directory `level1` containing subdirectory `level2`, use the command:

```
awplus# mkdir level1
awplus# mkdir level1/level2
awplus# rmdir force level1
```

To remove a directory called `test` from the top level of the Flash filesystem, in stack member 3, use the command:

```
awplus# rmdir awplus-3/flash:/test
```

Note that you must specify the filesystem, ("flash:" in this example).

Related Commands
cd
dir
mkdir
pwd

show autoboot

This command displays the Autoboot configuration and status.

Syntax `show autoboot`

Mode Privileged Exec

Example To show the Autoboot configuration and status, use the command:

```
awplus# show autoboot
```

Output **Figure 7-5: Example output from the show autoboot command**

```
awplus#show autoboot
Autoboot configuration
------------------------------------------------------------------------------
Autoboot status                  : enabled
USB file autoboot.txt exists     : yes

Restore information on USB
Autoboot enable in autoboot.txt  : yes
Restore release file             : x510-5.4.4-3.5.rel (file exists)
Restore configuration file       : network_1.cfg (file exists)
```

Figure 7-6: Example output from the show autoboot command when an external media source is not present

```
awplus#show autoboot
Autoboot configuration
------------------------------------------------------------------------------
Autoboot status                  : enabled
External media source            : USB not found.
```

Related Commands **autoboot enable**
create autoboot
show boot

show boot

This command displays the current boot configuration.

Syntax `show boot`

Mode Privileged Exec

Example To show the current boot configuration, use the command:

```
awplus# show boot
```

Output **Figure 7-7: Example output from the show boot command with the current boot config set on a USB storage device**

```
awplus#show boot
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : usb:/SBx81CFC960-5.4.4-1.2.rel
Backup  boot image : flash:/SBx81CFC960-5.4.4-1.1.rel
Default boot config: flash:/default.cfg
Current boot config: usb:/my.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file not found)
Autoboot status    : enabled
```

Figure 7-8: Example output from the show boot command

```
awplus#show boot
Boot configuration
-----------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : flash:/SBx81CFC960-5.4.4-1.2.rel
Backup  boot image : flash:/SBx81CFC960-5.4.4-1.1.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/my.cfg (file exists)
Backup  boot config: flash:/backup.cfg (file not found)
Autoboot status    : enabled
```

Table 7-1: Parameters in the output of the show boot command

Parameter	Description
`Current software`	The current software release that the device is using.
`Current boot image`	The boot image currently configured for use during the next boot cycle.
`Backup boot image`	The boot image to use during the next boot cycle if the device cannot load the main image.
`Default boot config`	The default startup configuration file. The device loads this configuration script if no file is set as the startup-config file.
`Current boot config`	The configuration file currently configured as the startup-config file. The device loads this configuration file during the next boot cycle if this file exists.

Parameter	Description
`Backup boot config`	The configuration file to use during the next boot cycle if the main configuration file cannot be loaded.
`Autoboot status`	The status of the Autoboot feature; either enabled or disabled.

Related Commands

autoboot enable
boot config-file backup
boot system backup
show autoboot

show file

This command displays the contents of a specified file.

Syntax `show file {<filename>|<url>}`

Parameter	Description
<filename>	Name of a file on the local Flash filesystem.
<url>	URL of a file.

Mode Privileged Exec

Example To display the contents of the file `oldconfig.cfg`, which is in the current directory, use the command:

```
awplus# show file oldconfig.cfg
```

Related Commands
edit
edit (URL)
show file systems

show file systems

This command lists the filesystems and their utilization information where appropriate.

If this command is entered on the stack master, it will list the filesystems for all the stack members. A stack member heading is displayed to distinguish the different lists shown for each stack member.

Syntax `show file systems`

Mode Privileged Exec

Examples To display the filesystems for either a standalone device, or a complete stack, use the command:

```
awplus# show file systems
```

Output **Figure 7-9: Example output from the show file systems command**

```
awplus#show file systems
 Size(b)  Free(b)  Type   Flags  Prefixes   S/D/V   Lcl/Ntwk  Avail
-------------------------------------------------------------------
  63.0M    29.4M   flash     rw  flash:     static  local      Y
      -        -   system    rw  system:    virtual local      -
  10.0M     9.9M   debug     rw  debug:     static  local      Y
 499.0K   404.0K   nvs       rw  nvs:       static  local      Y
      -        -   usbstick  rw  usb:       dynamic local      N
      -        -   tftp      rw  tftp:      -       network    -
      -        -   scp       rw  scp:       -       network    -
      -        -   sftp      ro  sftp:      -       network    -
      -        -   http      ro  http:      -       network    -
      -        -   rsync     rw  rsync:     -       network    -
```

Table 7-2: Parameters in the output of the show file systems command

Parameter	Description
`Size (B)` `Available`	The total memory available to this filesystem. The units are given after the value and are M for Megabytes or k for kilobytes.
`Free (B)`	The total memory free within this filesystem. The units are given after the value and are M for Megabytes or k for kilobytes.
`Type`	The memory type used for this filesystem: flash, system, nvs, usbstick, tftp, scp, sftp, or http.
`Flags`	The file setting options: rw (read write), ro (read only).
`Prefixes`	The prefixes used when entering commands to access the filesystems: flash, system, nvs, usb, tftp, scp, sftp, or http.
`S/V/D`	The memory type: static, virtual, dynamic.
`Lcl / Ntwk`	Whether the memory is located locally or via a network connection.
`Avail`	Whether the memory is accessible: Y (yes), N (no), - (not appropriate)

Related Commands **edit**
edit (URL)
show file

show running-config

This command displays the current configuration of the device. The output includes all non-default configuration; default settings are not displayed.

You can control the output in any one of the following ways:

- To display only lines that contain a particular word, enter **| include word** after the command
- To start the display at the first line that contains a particular word, enter **| begin word** after the command
- To save the output to a file, enter **> filename** after the command

For more information, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show running-config`

Mode Privileged Exec and Global Configuration

Example To display the current dynamic configuration of your device, use the command:

```
awplus# show running-config
```

Output **Figure 7-10: Example output from the show running-config command**

```
awplus#show running-config
!
service password-encryption
!
hostname MyNode
!
no banner motd
!
username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0
!
no service ssh
!
autoboot enable
!
service telnet
!
service http
!
no clock timezone
!
snmp-server
snmp-server contact Documentation Area
snmp-server location New Zealand
!
aaa authentication enable default local
aaa authentication login default local
!
ip domain-lookup
!
no service dhcp-server
!
no ip multicast-routing
!
spanning-tree mode rstp
!
no spanning-tree rstp enable
!
switch 1 provision x510-28
!
vlan database
 vlan 2-15 state enable
!
interface port1.0.1-1.0.6
 switchport
 switchport mode access
!
interface port1.0.25-1.0.26
 switchport
 switchport mode access
 switchport access vlan 14
!
interface vlan1
 ip address 192.168.1.1/24
 ipv6 enable
 ipv6 mld
!
interface vlan12
 ip address 192.168.3.1/24
!
ipv6 forwarding
!
line con 0
line vty 0 4
!
end
```

Related Commands **copy running-config**
show running-config access-list

show running-config access-list

Use this command to show the running system status and configuration details for access-list.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config access-list
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system status and configuration details for access-list, use the command:

```
awplus# show running-config access-list
```

Output

Figure 7-11: Example output from the show running-config access-list command

```
!
access-list abc remark annai
access-list abc deny any
access-list abd deny any
!
```

Related Commands

copy running-config
show running-config

show running-config as-path access-list

Use this command to show the running system status and configuration details for as-path access-list.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config as-path access-list
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system status and configuration details for as-path access-list, use the command:

```
awplus# show running-config as-path access-list
```

Output

Figure 7-12: Example output from the show running-config as-path access-list command

```
!
ip as-path access-list wer permit knsmk
!
```

Related Commands

copy running-config
show running-config

show running-config dhcp

Use this command to display the running configuration for DHCP server, DHCP snooping, and DHCP relay.

Syntax `show running-config dhcp`

Mode Privileged Exec and Global Configuration

Example To display to display the running configuration for DHCP server, DHCP snooping, and DHCP relay:

```
awplus# show running-config dhcp
```

Output **Figure 7-13: Example output from the show running-config dhcp command**

```
!
#show running-config dhcp
no service dhcp-server
!
service dhcp-snooping
!
interface port1.0.1
 ip dhcp snooping trust
!
interface port1.0.3
 ip dhcp snooping max-bindings 25
 access-group dhcpsnooping
!
interface port1.0.4
 ip dhcp snooping max-bindings 25
 access-group dhcpsnooping
!
interface po1
 ip dhcp snooping max-bindings 25
 arp security violation log
!
interface sa1
 ip dhcp snooping max-bindings 25
 access-group dhcpsnooping
 arp security violation log
!
interface vlan100
 ip dhcp snooping
 arp security
!
interface vlan200
 ip dhcp snooping
 arp security
!
```

Related Commands copy running-config
show running-config

show running-config full

Use this command to show the complete status and configuration of the running system.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show running-config full`

Mode Privileged Exec and Global Configuration

Example To display the complete status and configuration of the running system, use the command:

```
awplus# show running-config full
```

Related Commands **copy running-config**
show running-config

show running-config interface

This command displays the current configuration of one or more interfaces on the switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config interface [<interface-list>] [dot1x|ip igmp|
   ip multicast|ip pim dense-mode|ip pim sparse-mode|ipv6 rip|lacp|
   mstp|ospf|rip|rstp|stp]
```

Parameter	Description
<interface-list>	The interfaces or ports to display information about. An interface-list can be: ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen, e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above, e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`dot1x`	Displays running configuration for 802.1X port authentication for the specified interfaces.
`lacp`	Displays running configuration for LACP (Link Aggregation Control Protocol) for the specified interfaces.
`ip igmp`	Displays running configuration for IGMP (Internet Group Management Protocol) for the specified interfaces.
`ip multicast`	Displays running configuration for general multicast settings for the specified interfaces.
`ip pim sparse-mode`	Displays running configuration for PIM-SM (Protocol Independent Multicast - Sparse Mode) for the specified interfaces.
`ip pim dense-mode`	Displays running configuration for PIM-DM (Protocol Independent Multicasting - Dense Mode) for the specified interfaces.
`mstp`	Displays running configuration for MSTP (Multiple Spanning Tree Protocol) for the specified interfaces.
`ospf`	Displays running configuration for OSPF (Open Shortest Path First) for the specified interfaces.
`rip`	Displays running configuration for RIP (Routing Information Protocol) for the specified interfaces.
`ipv6 rip`	Displays running configuration for RIPng (RIP for IPv6) for the specified interfaces.

Parameter	Description
`rstp`	Displays running configuration for RSTP (Rapid Spanning Tree Protocol) for the specified interfaces.
`stp`	Displays running configuration for STP (Spanning Tree Protocol) for the specified interfaces.

Mode Privileged Exec and Global Configuration

Examples To display the current running configuration of your switch for ports 1 to 4, use the command:

```
awplus# show running-config interface port1.0.1-port1.0.4
```

To display the current running configuration of a switch for VLAN 1, use the command:

```
awplus# show running-config interface vlan1
```

To display the current running configuration of a switch for VLANs 1 and 3-5, use the command:

```
awplus# show running-config interface vlan1,vlan3-vlan5
```

To display the current OSPF configuration of your switch for ports 1 to 24, use the command:

```
awplus# show running-config interface port1.0.1-port1.0.6
       ospf
```

Output **Figure 7-14: Example output from a show running-config interface port1.0.2 command**

```
awplus#sh running-config interface port1.0.2
!
interface port1.0.2
 switchport
 switchport mode access
!
```

Figure 7-15: Example output from the show running-config interface command

```
awplus#show running-config interface
interface port1.0.1-1.0.6
 switchport
 switchport mode access
!
interface port1.0.25-1.0.26
 switchport
 switchport mode access
 switchport access vlan 14
!
interface port1.0.27-1.0.28
 switchport
 switchport mode access
 switchport access vlan 15
!
interface vlan1
 ip address 192.168.1.1/24
 ipv6 enable
 ipv6 mld
!
interface vlan12
 ip address 192.168.3.1/24
!
interface vlan13
 ip address 192.168.2.1/24
```

Related Commands

copy running-config
show running-config

show running-config ip pim dense-mode

Use this command to show the running system status and configuration details for PIM-DM.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config ip pim dense-mode
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system status and configuration details for PIM-DM, use the command:

```
awplus# show running-config ip pim dense-mode
```

Output

Figure 7-16: Example output from the show running-config ip pim dense-mode command

```
!
ip pim spt-threshold
ip pim accept-register list 1
!
```

Related Commands

copy running-config
show running-config

show running-config ip pim sparse-mode

Use this command to show the running system status and configuration details for PIM-SM.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config ip pim sparse-mode
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system status and configuration details for PIM-SM, use the command:

```
awplus# show running-config ip pim sparse-mode
```

Output

Figure 7-17: Example output from the show running-config ip pim sparse-mode command

```
!
ip pim spt-threshold
ip pim accept-register list 1
!
```

Related Commands

copy running-config
show running-config

show running-config ip route

Use this command to show the running system static IPv4 route configuration.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config ip route
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system static IPv4 route configuration, use the command:

```
awplus# show running-config ip route
```

Output

Figure 7-18: Example output from the show running-config ip route command

```
!
ip route 3.3.3.3/32 vlan3
ip route 3.3.3.3/32 vlan2
!
```

Related Commands

copy running-config
show running-config

show running-config ipv6 access-list

Use this command to show the running system status and configuration for IPv6 ACLs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show running-config ipv6 access-list`

Mode Privileged Exec and Global Configuration

Example To display the running system status and configuration for IPv6 ACLs, use the command:

```
awplus# show running-config ipv6 access-list
```

Output **Figure 7-19: Example output from the show running-config ipv6 access-list command**

```
!
ipv6 access-list abc permit any
!
```

Related Commands **copy running-config**
show running-config

show running-config ipv6 mroute

Use this command to show the running system IPv6 multicast route configuration.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config ipv6 mroute
```

Mode

Privileged Exec and Global Configuration

Example

To display he running system IPv6 multicast route configuration, use the command:

```
awplus# show running-config ipv6 mroute
```

Output

Figure 7-20: Example output from the show running-config ipv6 mroute command

```
!
ipv6 route 3e11::/64 lo
ipv6 route 3e11::/64 vlan2
ipv6 route fe80::/64 vlan3
!
```

Related Commands

copy running-config
show running-config

show running-config ipv6 prefix-list

Use this command to show the running system status and configuration details for IPv6 prefix lists.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config ipv6 prefix-list
```

Mode

Privileged Exec and Global Configuration

Example

To display show the running system status and configuration details for IPv6 prefix lists, use the command:

```
awplus# show running-config ipv6 prefix-list
```

Output

Figure 7-21: Example output from the show running-config ipv6 prefix-list command

```
!
ipv6 prefix-list sde seq 5 permit any
!
```

Related Commands

copy running-config
show running-config

show running-config ipv6 route

Use this command to show the running system static IPv6 route configuration.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show running-config ipv6 route`

Mode Privileged Exec and Global Configuration

Example To display the running system static IPv6 route configuration, use the command:

```
awplus# show running-config ipv6 route
```

Output **Figure 7-22: Example output from the show running-config ipv6 route command**

```
!
ipv6 route 3e11::/64 lo
ipv6 route 3e11::/64 vlan2
ipv6 route fe80::/64 vlan3
!
```

Related Commands copy running-config
show running-config

show running-config key chain

Use this command to show the running system key-chain related configuration.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show running-config key chain`

Mode Privileged Exec and Global Configuration

Example To display the running system key-chain related configuration, use the command:

```
awplus# show running-config key chain
```

Output **Figure 7-23: Example output from the show running-config key chain command**

```
!
key chain 12
key 2
key-string 234
!
key chain 123
key 3
key-string 345
!
```

Related Commands **copy running-config**
show running-config

show running-config lldp

This command shows the current running configuration of LLDP.

Syntax `show running-config lldp`

Mode Privileged Exec and Global Configuration

Example To display the current configuration of LLDP, use the command:

```
awplus# show running-config lldp
```

Output **Figure 7-24: Example output from the show running-config lldp command**

```
awplus#show running-config lldp

lldp notification-interval 10
lldp timer 20
!
interface port1.0.1
 lldp notifications
 lldp tlv-select port-description
 lldp tlv-select system-name
 lldp tlv-select system-description
 lldp tlv-select management-address
 lldp transmit receive
```

Related Commands **show lldp**
show lldp interface

show running-config power-inline

Use this command to show the Power over Ethernet (PoE) running system status and configuration details. The PoE usage-threshold percentage as specified by the **power-inline usage-threshold** command is displayed in the **running-config** using this command.

See **Chapter 24, Power over Ethernet Introduction** and **Chapter 25, Power over Ethernet Commands** for more information about PoE.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config power-inline
```

Mode

Privileged Exec and Global Configuration

Example

To display the PoE running system status and configuration details, use the command:

```
awplus# show running-config power-inline
```

Output

Figure 7-25: Example output from the show running-config power-inline command

```
!
power-inline usage-threshold 90
!
```

Related Commands

power-inline usage-threshold
show power-inline

show running-config prefix-list

Use this command to show the running system status and configuration details for prefix-list.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config prefix-list
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system status and configuration details for prefix-list, use the command:

```
awplus# show running-config prefix-list
```

Output

Figure 7-26: Example output from the show running-config prefix-list command

```
!
ip prefix-list abc seq 5 permit any
ip prefix-list as description annai
ip prefix-list wer seq 45 permit any
!
```

Related Commands

copy running-config
show running-config

show running-config route-map

Use this command to show the running system status and configuration details for route-map.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config route-map
```

Mode

Privileged Exec and Global Configuration

Example

To display the running system status and configuration details for route-map, use the command:

```
awplus# show running-config route-map
```

Output

Figure 7-27: Example output from the show running-config route-map command

```
!
route-map abc deny 2
match community 2
!
route-map abc permit 3
match route-type external type-2
set metric-type type-1
!
```

Related Commands

copy running-config
show running-config

show running-config router

Use the show running-config router command to display the current running configuration for a given router.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show running-config router <protocol>`

Parameter	Description
<protocol>	ospf | rip |ipv6 rip | vrrp
`bgp`	Border Gateway Protocol (BGP)
`ospf`	Open Shortest Path First (OSPF)
`rip`	Routing Information Protocol (RIP)
`ipv6 rip`	IPv6 RIP
`vrrp`	Virtual Redundancy Routing Protocol (VRRP)

Mode Privileged Exec and Global Configuration

Example To display the current running configuration for a given router, use the command:

```
awplus# show running-config router ospf
```

Output **Figure 7-28: Example output from the show running-config router command**

```
!
router ospf
 network 192.168.1.0/24 area 0.0.0.0
 network 192.168.3.0/24 area 0.0.0.0
!
```

Related Commands **copy running-config**
show running-config

show running-config router-id

Use this command to show the running system global router ID configuration.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show running-config router-id`

Mode Privileged Exec and Global Configuration

Example To display the running system global router ID configuration, use the command:

```
awplus# show running-config router-id
```

Output **Figure 7-29: Example output from the show running-config router-id command**

```
!
router-id 3.3.3.3
!
```

Related Commands **copy running-config**
show running-config

show running-config security-password

This command displays the configuration settings for the various security-password rules. If a default parameter is used for a security-password rule, therefore disabling that rule, no output is displayed for that feature.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show running-config security-password
```

Mode

Privileged Exec and Global Configuration

Example

To display the current security-password rule settings in the running-config, use the command:

```
awplus# show running-config security-password
```

Output

Figure 7-30: Example output from the show running-config security-password command

```
security-password minimum-length 8
security-password minimum-categories 3
security-password history 4
security-password lifetime 30
security-password warning 3
security-password forced-change
```

Related Commands

show security-password configuration
show security-password user

show startup-config

This command displays the contents of the start-up configuration file, which is the file that the device runs on start-up.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show startup-config`

Mode Privileged Exec

Example To display the contents of the current start-up configuration file, use the command:

```
awplus# show startup-config
```

Output **Figure 7-31: Example output from the show startup-config command**

```
awplus#show startup-config
!
service password-encryption
!
username manager privilege 15 password 8 $1$bJoVec4D$JwOJGPr7YqoExA0GVasdE0
!
no service ssh
!
service telnet
!
service http
!
no clock timezone
.
.
.
line con 0
line vty 0 4
!
end
```

Related Commands
boot config-file backup
copy running-config
copy startup-config
erase startup-config
show boot

show version

This command displays the version number and copyright details of the current AlliedWare Plus™ OS your device is running.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show version`

Mode User Exec and Privileged Exec

Example To display the version details of your currently installed software, use the command:

```
awplus# show version
```

Output **Figure 7-32: Example output from the show version command**

```
awplus#show version

AlliedWare Plus (TM) 5.4.3 19/11/12 13:22:32

Build name : x510-5.4.4-3.5.rel
Build date : Fri Jun 6 13:22:32 NZST 2014
Build type : RELEASE
 NET-SNMP SNMP agent software
   (c) 1996, 1998-2000 The Regents of the University of California.
       All rights reserved;
   (c) 2001-2003, Networks Associates Technology, Inc. All rights reserved.
   (c) 2001-2003, Cambridge Broadband Ltd. All rights reserved.
   (c) 2003, Sun Microsystems, Inc. All rights reserved.
   (c) 2003-2006, Sparta, Inc. All rights reserved.
   (c) 2004, Cisco, Inc and Information Network
       Center of Beijing University of Posts and Telecommunications.
       All rights reserved.
 RSA Data Security, Inc. MD5 Message-Digest Algorithm
   (c) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved.
 OpenSSL Library
   Copyright (C) 1998-2011 The OpenSSL Project. All rights reserved.
 Original SSLeay License
   Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com).
 sFlow(R) Agent Software
   Copyright (c) 2002-2006 InMon Corp.
 DHCP Library
   Copyright (c) 2004-2012 by Internet Systems Consortium, Inc. ("ISC")
   Copyright (c) 1995-2003 by Internet Software Consortium.
 DHCP Bind
   Copyright (c) 2005 - 2008, Holger Zuleger HZnet. All rights reserved.
 Application Interface Specification Framework
   Copyright (c) 2002-2004 MontaVista Software, Inc;
   Copyright (c) 2005-2010 Red Hat, Inc.
 Hardware Platform Interface Library
   Copyright (c) 2004 by Intel Corp.
   Copyright (C) IBM Corp. 2004-2008.
 Corosync Cluster Engine
   Copyright (c) 2002-2004 MontaVista Software, Inc. All rights reserved.
   Copyright (c) 2005-2010 Red Hat, Inc. File Utility Library
   Copyright (c) Ian F. Darwin 1986-1987, 1989-1992, 1994-1995.
   Software written by Ian F. Darwin and others;
   maintained 1994- Christos Zoulas.
 ProL2TP
   Copyright Katalix Systems Ltd, 2010, 2011.
   All rights reserved.

Portions of this product are covered by the GNU GPL, source code may be
downloaded from: http://www.alliedtelesis.co.nz/support/gpl/awp.html
```

Related Commands **boot system backup**
show boot

write file

This command copies the running-config into the file that is set as the current startup-config file. This command is a synonym of the **write memory** and **copy running-config startup-config** commands.

Syntax `write [file]`

Mode Privileged Exec

Example To write configuration data to the start-up configuration file, use the command:

```
awplus# write file
```

Related Commands **copy running-config**
write memory
show running-config

write memory

This command copies the running-config into the file that is set as the current startup-config file. This command is a synonym of the **write file** and **copy running-config startup-config** commands.

Syntax `write [memory]`

Mode Privileged Exec

Example To write configuration data to the start-up configuration file, use the command:

```
awplus# write memory
```

Related Commands **copy running-config**
write file
show running-config

write terminal

This command displays the current configuration of the device. This command is a synonym of the **show running-config** command.

Syntax `write terminal`

Mode Privileged Exec

Example To display the current configuration of your device, use the command:

```
awplus# write terminal
```

Related Commands **show running-config**

Chapter 8: Licensing Introduction and Configuration

Introduction 8.2
About Licensing VCStack 8.2
Licensing Terminology 8.2
Applying a New Feature License on a Stand-Alone Switch 8.4
VCStack Licensing Configuration 8.5
Adding a New Switch to a VCStack 8.5
Adding a New Feature License to a VCStack 8.6

Introduction

Licensing

Allied Telesis devices running AlliedWare Plus software can have base and feature software licensing. These licenses enable default and additional software features, respectively.

A base license enables default software features. The default features are determined by region. The region is shown as the “Board region” in console output.

Feature license keys enable additional software features that are not included with a base license, and are used to run advanced features such as Layer 3 routing and IPv6 protocols.

See **Licensing Terminology** for definitions of licensing terms used through this chapter.

About Licensing VCStack

In order for a stack to form, each switch in the stack needs to be running the same version of AlliedWare Plus software.

It is important for the integrity of the stack that all stack members have the same feature licenses, although a stack can still form if its members have differing feature licenses. Note that in the case of master failover, if a stack member that lacks a certain feature license becomes the new stack master, the feature will not be exhibited by the stack even if all other stack members are licensed for that feature.

If you introduce a new stack member that lacks a certain feature license possessed by the other stack members, a warning message will be generated to inform you that the stack may not operate to its full potential.

Switches with different OEM Territories may have different base feature licenses, so while a stack may form, a warning message may be generated to inform you that the features do not match on all stack members.

For step-by-step instructions for licensing on stacked switches, see **VCStack Licensing Configuration**.

Licensing Terminology

See the table below for descriptions of licensing terms used through this chapter.

Term	Description
Activation	A process by which a feature is enabled. A set of base features is available on a given device. Activated devices do not show console or log messages requesting activation. Unlicensed devices display console messages requesting activation and are unsupported.
Feature	A software feature or protocol that is enabled with a valid software license. A software feature license key enables a feature on a device. A license is valid if it is appropriate for a software feature or protocol. For example, an IPv6 feature license will only enable IPv6 not ESPR.
License	A software object that enables features on a device. A license is a combination of an identifying license label and a license key.

Term	Description
License Key	An encrypted string of characters associated with a particular license that when added to a device enables a feature on the device.
License Label	A descriptive name associated with a license. This name may be chosen by a user, though license generation provides a default label for each license. This may also be called a license name.

Applying a New Feature License on a Stand-Alone Switch

Feature licenses are applied with the **license** command and validated with the **show license** and **show license brief** commands. Follow these steps to apply feature licenses:

- **Purchase a feature license for a switch**
- **Apply a feature license on a switch**
- **Confirm feature license application on a switch**

Step 1: Purchase a feature license for a switch

Purchase a feature license from your authorized distributor or reseller.

See the AlliedWare Plus datasheet for a list of licenses available by device.

Step 2: Apply a feature license on a switch

Use the **license** command to apply a feature license to your switch.

Note that if the feature license contains a license for a protocol, then that protocol will restart. This action may result in the loss of network traffic. We advise that you should only install licenses during scheduled maintenance for devices operating in a live environment. When you add a feature license you are warned on the console before that feature restarts.

Step 3: Confirm feature license application on a switch

AlliedWare Plus switches have a base feature license applied, which is shown in **show license** and **show license brief** command output. The output from these commands will also show any other licenses that currently apply to the switch.

Figure 8-1: Example show license brief command output showing a base feature license on a stand-alone switch

```
awplus#show license brief

Board region: Global
Software Licenses
-------------------------------------------------------------------------
1     Base License    1            Base License
      Full                       N/A

Current enabled features for displayed licenses:
 EPSR-MASTER, IPv6Basic, LAG-FULL, MLDSnoop, OSPF-64, RADIUS-100, RIP, VRRP
```

VCStack Licensing Configuration

This section describes the licensing configuration for stacked switches. It is important for the integrity of the stack that all stack members have the same feature licenses, although a stack can still form if its members have differing feature licenses. Note that in the case of master failover, if a stack member that lacks a certain feature license becomes the new stack master, that feature will not be exhibited by the stack even if all other stack members are licensed for that feature.

If you introduce a new stack member that lacks a certain feature license possessed by the other stack members, a warning message will be generated to inform you that the stack may not operate to its full potential.

See the **VCStack Introduction** chapter for a stacking overview, and **Stack Formation** for more information on the processes a stack goes through when it forms.

Note that to maintain consistent behavior across the stack, all member switches must have the same feature licenses enabled.

Adding a New Switch to a VCStack

Prior to the 5.4.4-1.x software release, stacks could form only if all members of the stack had the same features licensed. In practice, when a new switch was to be added to an existing stack, it first had to have feature licenses applied to it before it could be connected to the stack.

Since 5.4.4-1.x, this is no longer the case — the new device can be connected to the existing stack and it will join without problem.

Therefore, for information about connecting a new device to a stack, see your switch's Installation Guide or **Chapter 110, VCStack Introduction** of this Software Reference. If there is a licensing mismatch, a warning message will be generated to inform you that the stack may not operate to its full potential. In this case you should follow the instructions in **"Adding a New Feature License to a VCStack" on page 8.6** to correct the problem.

Adding a New Feature License to a VCStack

Follow this section to add a new feature license to all members of a VCStack, including any newly installed members that may not already have the required license(s). If you need to license a single stack member then follow the steps in **Adding a New Switch to a VCStack**.

Perform the following tasks to add a new feature license to a VCStack:

- **Apply a feature license to all members of a VCStack**
- **Reboot the stack (optional)**

Step 1: Apply a feature license to all members of a VCStack

Use the **license** command to apply the required feature license to the VCStack (see **Figure 8-2**). In a stacked configuration, the **license** command will add a license to all stack members and the **no license** command will remove a license from all stack members.

Previously, feature licenses were applied to each stack member using the **license member** command. Since version 5.4.4, the **license** command applies feature licenses to all stack members. You do not need to use the **remote-login** command to log in to stack members from a master.

If a stack member already has the new license installed, the **license** command will be ignored for that stack member and no action will take place. In this way, a license will only be applied to devices that need it, allowing an individual stack member to easily be brought into sync with the other members of the stack.

Figure 8-2: Example license command entry to add a feature license to a stack

```
awplus#license IPv6 Qd0NvZJ8DutyLAYbsM8pCpY1d8Ho9mzygweBp+paBqVu7By1bTZ+Jipo57
A restart of affected modules may be required.
Would you like to continue? (y/n): y

Stack member 1 installed 1 license

1 license installed.
```

Step 2: Reboot the stack (optional)

You may choose to reboot the stack after the new license has installed, by using either the **reboot** command or the **reload** command. Note that rebooting is optional, but it is good practice as it enables you to check the stack. If you choose not to reboot, the feature will restart on the stack.

Figure 8-3: Example reboot command output showing a valid license console message

```
reboot system? (y/n): y

URGENT: broadcast message:
System going down IMMEDIATELY!

        ... Rebooting at user request ...


Flushing file system buffers...
Unmounting any remaining filesystems...
Restarting system.

Bootloader 2.0.13 loaded
Press <Ctrl+B> for the Boot Menu

Verifying release... OK
Booting...
Starting base/first...                                  [  OK  ]
Mounting virtual filesystems...                         [  OK  ]

           _______________   ____
        /\ \             / /______\
      /   \ \_      __/ /| ______ |
    /      \ |     |  /  | ______ |
  /         \ \   / /     \ ____ /
/______/\____\ \/ /____________/


Received event network.initialized
Received event standalone

10:17:31 awplus IMI[1718]: SFL: The current software is licensed. Exiting
unlicensed mode.
12:11:23 awplus-2 VCS[1865]: Member 1 (eccd.6d48.e560) has joined the stack
12:11:24 awplus-2 VCS[1865]: Member 3 (eccd.6d5e.2614) has joined the stack
12:11:26 awplus-2 VCS[1865]: Member 1 (eccd.6d48.e560) has become the Active
Master

Received event network.activated

Loading default configuration
.

done!
Received event network.configured

awplus login: manager
Password:
awplus>
```

Chapter 9: Licensing Commands

Command List 9.2
license 9.2
license member (deleted) 9.3
show license 9.4
show license brief 9.6
show license member 9.8
show license brief member 9.10

Command List

This chapter provides an alphabetical reference for each of the License commands.

license

This command activates the licensed software feature set on a standalone switch, or a stack of switches.

Use the **no** variant of this command to deactivate the licensed software feature set on a standalone switch, or a stack of switches.

For feature licenses, contact your authorized distributor or reseller. If a license key expires or is incorrect so the license key is invalid, then some software features will be unavailable.

Note

See the AlliedWare Plus™ datasheet for a list of current feature licenses available by product, and the AlliedWare Plus™ How To notes for information on obtaining them. Purchase licenses from your authorized dealer or reseller.

Only install feature licenses during scheduled maintenance for any devices in a live environment. For example, if a feature license includes EPSR, EPSR is restarted with a temporary loss of EPSR network traffic.

Syntax

```
license <label> <key>
no license <label>
```

Parameter	Description
<label>	A name for the feature license. To determine names already in use, use the **show license** command. This can be the default name supplied for the feature, or a renamed feature name.
<key>	The encrypted license key to enable a set of software features.

Mode Privileged Exec

Usage You can change the license label using this command to make it specific to you when you initially add a license. Once a license is added, any change to the license label first requires removal of the license before adding a license again with a new license label.

The default feature license labels are issued along with encrypted license keys by e-mail for you to apply using this command to activate features. You can change default feature license labels, but they must be 15 characters or less to be accepted with the issued keys.

For example, you may want to change the label of the premium license to “premium-license”. You can check your new license label by using the **show license** command.

In a stacked configuration, the **license** command will add a license to all stack members and the **no license** command will remove a license from all stack members.

You can add a license to a specified stack member after first using the **remote-login** command from the stack master. Adding or deleting licenses on individual switches can cause different members of the stack to have different features enabled, which may cause the stack to fail to operate correctly. Unbalanced stack members will not form a stack. Stack members require the same feature licenses to be balanced.

If you add a feature license you will be prompted at the console that the feature needs to restart. For example, if the feature license contains a license for the EPSR protocol, then that protocol will restart. This action may result in the loss of network traffic. Only install licenses in scheduled maintenance periods for devices in a live environment.

Note that operating the following features require an x510 Premium License (AT-FL-x510-01):

- RIP
- OSPF
- PIMv4-SM, DM, & SSM
- EPSR Master
- VLAN Double Tagging (Q in Q)
- RIPng
- OSPFv3
- MLDv1 & v2
- PIMv6-SM

Examples To activate the license `name1` with the key `12345678ABCDE123456789ABCDE`, use the command:

```
awplus# license name1 12345678ABCDE123456789ABCDE
```

To deactivate the license `name1`, use the command:

```
awplus# no license name1
```

Output **Figure 9-1: Example of a license command entry to remove a feature license**

```
awplus#no license IPv6
Stack member 1: Removal of "IPv6" will disable the following features:
 IPv6

INFO: Uninstalling license key will disable the affected modules immediately.
Would you like to continue? (y/n): y

Stack member 1 removed 1 license

1 license removed.
```

Validation Command **show license**

license member (deleted)

This command has been deleted. Use the **license** command to apply licenses to members.

In a stacked configuration, the **license** command will add a license to all stack members and the **no license** command will remove a license from all stack members.

show license

This command displays information about a specific software feature license, or all enabled software feature licenses on the device.

Syntax `show license [feature] [<label>|index <index-number>]`

Parameter	Description
`feature`	Only display license information for any applied feature licenses.
`<label>`	The license name of the software feature to show information about. The license name can be used instead of the index number to identify a specific license.
`index <index-number>`	The index number of the software feature license to show information about. The index number can be used instead of the license name to identify a specific license.

Mode User Exec and Privileged Exec

Usage In a stacked configuration, use the **show license member** command to display license information about either a specific licensed software feature, or all software features enabled on a specific stack member or on all stack members.

Note that the **show license** command will display licenses applied to a stack master only.

Examples To display full information about all enabled licenses, use the command:

```
awplus# show license
```

To display full information about the licenses with index number 1, use the command:

```
awplus# show license index 1
```

Output **Figure 9-2: Example output from the show license command showing a base license with index 1**

```
awplus#show license
Board region: Global
Software Licenses
------------------------------------------------------------------------
Index                          : 1
License name                   : Base License
Customer name                  : Base License
Quantity of licenses           : 1
Type of license                : Full
License issue date             : 12-Jul-2014
License expiry date            : N/A
Features included              : IPv6Basic, LAG-FULL, MLDSnoop, RADIUS-100, VCS,
                                 VRRP
```

Table 9-1: Parameters in the output of the show license command

Parameter	Description
`Board region`	Name of the region for the Base License features.
`Index`	Index identifying entry. The index is assigned automatically by the software. It is not configured.
`License name`	Name of the license key bundle (case-sensitive).
`Customer name`	Customer name.
`Quantity of licenses`	Quantity of licensed installations.
`Type of license`	Full or Trial.
`License issue date`	Date the license was generated.
`License expiry date`	Expiry date for trial license.
`Features included`	List of features included in the feature license.

Related Commands

license
show license brief
show license member

show license brief

This command displays information about a specific software feature license, or all enabled software feature licenses on the device.

In a stacked configuration, use the **show license member** command to display license information about either a specific licensed software feature, or all software feature licenses enabled on a specific stack member or on all stack members.

Syntax `show license [feature] [<label>|index <index-number>] brief`

Parameter	Description
`feature`	Only display license information for any applied feature licenses.
`<label>`	The license name of the software feature to show information about. The license name can be used instead of the index number to identify a specific license.
`index <index-number>`	The index number of the software feature license to show information about. The index number can be used instead of the license name to identify a specific license.
`brief`	Displays a brief summary of feature license information.

Mode User Exec and Privileged Exec

Examples To display a brief summary of information about all feature licenses, use the command:

```
awplus# show license feature brief
```

Output **Figure 9-3: Example output from the show license brief command**

```
awplus#show license brief
Board region: Global
Software Licenses
-------------------------------------------------------------------------
Index License name    Quantity     Customer name
      Type            Version      Period
-------------------------------------------------------------------------
1     Base License    1            Base License
      Full                         N/A

Current enabled features for displayed licenses:
 IPv6Basic, LAG-FULL, MLDSnoop, RADIUS-100, VCS, VRRP
```

Table 9-2: Parameters in the output of the show license brief command

Parameter	Description
`Board region`	Name of the region for the Base License features.
`Index`	Index identifying entry. The index is assigned automatically by the software. It is not configured.
`License name`	Name of the license key bundle (case-sensitive).
`Quantity`	Quantity of licensed installations.
`Customer name`	Customer name.
`Type`	Full or Trial.
`Period`	Expiry date for trial license.
`Current enabled features for displayed licenses`	List of features included in the license.

Related Commands

license
show license
show license member

show license member

Use this command to display information about either a specific software license, or all software feature licenses enabled on either a specific stack member or all stack members.

Syntax `show license [<label>] member [<1-8>|all]`

Parameter	Description
<label>	The name of the license to show information about.
<1-8>	The ID of the stack member to show information about.
`all`	Display information about all stack members.

Mode User Exec and Privileged Exec

Usage Use the **show license member all** command to display full list output of all licenses per stack member.

Examples To display full information about all enabled licenses on all stack members, use the command:

```
awplus# show license member all
```

To display full information about all enabled licenses on stack member 2, use the command:

```
awplus# show license member 2
```

To display full information about the license `name1` on all stack members, use the command:

```
awplus# show license name1 member all
```

Output **Figure 9-4: Example output from the show license member command**

```
awplus#show license member all
Board region: Global
Software Feature Licenses
-------------------------------------------------------------------------
Index                          : 1
License name                   : Base License
Customer name                  : Base License
Quantity of licenses           : 1
Type of license                : Full
License issue date             : 12-Jul-2014
License expiry date            : N/A
Features included              : IPv6Basic, LAG-FULL, MLDSnoop, RADIUS-100, VCS,
                                 VRRP

Index                          : 2
License name                   : PIM Trial
Customer name                  : PIM Trial
Quantity of licenses           : 10
Type of license                : 30 day trial
License issue date             : 12-Jul-2014
License expiry date            : 12-Jul-2014
Features included              : PIM PIM-100
```

Table 9-3: Parameters in the output of the show license member command

Parameter	Description
`Board region`	Name of the region for the Base License features.
`Index`	Index identifying entry. The index is assigned automatically by the software. It is not configured.
`License name`	Name of the license key bundle (case-sensitive).
`Customer name`	Customer name.
`Quantity of licenses`	Quantity of licensed installations.
`Type of license`	Full or Trial.
`License issue date`	Date the license was generated.
`License expiry date`	Expiry date for trial license.
`Features included`	List of features included in the license.

Related Commands

license
show license
show license brief member

show license brief member

Use this command to display information about either a specific software license, or all software feature licenses enabled on either a specific stack member or all stack members.

Syntax `show license [<label>] brief member [<1-8>|all]`

Parameter	Description
<label>	The name of the license to show information about.
`brief`	Display a brief summary of license information.
<1-8>	The ID of the stack member to show information about.
`all`	Display information about all stack members.

Mode User Exec and Privileged Exec

Usage Use the **show license brief member all** command for brief table output of all licenses per stack member.

Examples To display a brief summary of information about all enabled licenses on stack member 2, use the command:

```
awplus# show license brief member 2
```

To display a brief summary about all enabled licenses on all stack members, use the command:

```
awplus# show license brief member all
```

To display a brief summary about the license `name1` on all stack members, use the command:

```
awplus# show license name1 brief member all
```

Output **Figure 9-5: Example output from the show license brief member command**

```
awplus#show license brief member 1
Board region: Global
Software Licenses
-------------------------------------------------------------------------
1     Base License    1            Base License
      Full                         N/A

Current enabled features for displayed licenses:
 IPv6Basic, LAG-FULL, MLDSnoop, RADIUS-100, VCS, VRRP
```

Table 9-4: Parameters in the output of the show license brief member command

Parameter	Description
`Board region`	Name of the region for the Base License features.
`Index`	Index identifying entry. The index is assigned automatically by the software. It is not configured.
`License name`	Name of the license key bundle (case-sensitive).
`Quantity`	Quantity of licensed installations.
`Customer name`	Customer name.
`Type`	Full or Trial.
`Period`	Expiry date for trial license.
`Current enabled features for displayed licenses`	List of features included in the license.

Related Commands

license
show license
show license member

Chapter 10: System Configuration and Monitoring Commands

Command List 10.2
banner exec 10.2
banner login (system) 10.4
banner motd 10.5
clock set 10.6
clock summer-time date 10.7
clock summer-time recurring 10.8
clock timezone 10.10
continuous-reboot-prevention 10.11
ecofriendly led 10.13
ecofriendly lpi 10.14
findme 10.16
hostname 10.18
max-fib-routes 10.20
max-static-routes 10.21
no debug all 10.22
reboot 10.23
reload 10.23
show clock 10.24
show continuous-reboot-prevention 10.25
show cpu 10.26
show cpu history 10.29
show debugging 10.31
show ecofriendly 10.32
show interface memory 10.34
show memory 10.36
show memory allocations 10.38
show memory history 10.39
show memory pools 10.41
show memory shared 10.42
show process 10.43
show reboot history 10.45
show router-id 10.46
show system 10.47
show system environment 10.48
show system interrupts 10.49
show system mac 10.50
show system pci device 10.51
show system pci tree 10.52
show system pluggable 10.53
show system pluggable detail 10.55
show system pluggable diagnostics 10.59
show system serialnumber 10.62
show tech-support 10.63
speed (asyn) 10.66
system territory (deprecated) 10.67
terminal monitor 10.67
undebug all 10.67

Command List

This chapter provides an alphabetical reference of commands for configuring and monitoring the system.

banner exec

This command configures the User Exec mode banner that is displayed on the console after you login. The **banner exec default** command restores the User Exec banner to the default banner. Use the **no banner exec** command to disable the User Exec banner and remove the default User Exec banner.

Syntax

```
banner exec <banner-text>
banner exec default
no banner exec
```

Default By default, the AlliedWare Plus™ version and build date is displayed at console login, such as:

```
AlliedWare Plus (TM) Version 5.4.4-2 03/31/14 00:44:25
```

Mode Global Configuration

Examples To configure a User Exec mode banner after login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner exec enable to move to Priv Exec mode
awplus(config)#exit
awplus#exit

awplus login: manager
Password:
enable to move to Priv Exec mode
awplus>
```

To restore the default User Exec mode banner after login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner exec default
awplus(config)#exit
awplus#exit

awplus login: manager
Password:

AlliedWare Plus (TM) Version 5.4.4-2 03/31/14 13:03:59


awplus>
```

To remove the User Exec mode banner after login, enter the following commands:

```
awplus#configure terminal
awplus(config)#no banner exec
awplus(config)#exit
awplus#exit

awplus login: manager
Password:
awplus>
```

Related Commands

banner login (system)
banner motd

banner login (system)

This command configures the login banner that is displayed on the console when you login. The login banner is displayed on all connected terminals. The login banner is displayed after the MOTD (Message-of-the-Day) banner and before the login username and password prompts.

Use the **no banner login** command to disable the login banner.

Syntax

```
banner login
no banner login
```

Default By default, no login banner is displayed at console login.

Mode Global Configuration

Examples To configure a login banner to be displayed when you login, enter the following commands:

```
awplus#configure terminal
awplus(config)#banner login
Type CNTL/D to finish.
authorised users only
awplus(config)#exit
awplus#exit

authorised users only
awplus login: manager
Password:

AlliedWare Plus (TM) Version 5.4.4-2 03/31/14 13:03:59


awplus>
```

To remove the login banner, enter the following commands:

```
awplus#configure terminal
awplus(config)#no banner login
awplus(config)#exit
awplus#exit

awplus login: manager
Password:

awplus>
```

Related Commands **banner exec**
banner motd

banner motd

Use this command to change the text MOTD (Message-of-the-Day) banner displayed before login. The MOTD banner is displayed on all connected terminals. The MOTD banner is useful for sending messages that affect all network users, for example, any imminent system shutdowns.

Use the **no** variant of this command to not display a text MOTD (Message-of-the-Day) banner on login.

Syntax

```
banner motd <motd-text>
no banner motd
```

Default By default, the switch displays the AlliedWare Plus™ OS version and build date before login.

Mode Global Configuration

Examples To configure a MOTD banner to be displayed when you login, enter the following commands:

```
awplus>enable
awplus#configure terminal
awplus(config)#banner motd system shutdown at 6pm
awplus(config)#exit
awplus#exit

system shutdown at 6pm
awplus login: manager
Password:

AlliedWare Plus (TM) Version 5.4.4-2 03/31/14 13:03:59
```

To remove the login banner, enter the following commands:

```
awplus>enable
awplus#configure terminal
awplus(config)#no banner motd
awplus(config)#exit
awplus#exit

awplus login: manager
Password:

AlliedWare Plus (TM) Version 5.4.4-2 03/31/14 13:03:59


awplus>
```

Related Commands banner exec
banner login (system)

clock set

This command sets the time and date for the system clock.

Syntax `clock set <hh:mm:ss> <day> <month> <year>`

Parameter	Description
<hh:mm:ss>	Local time in 24-hour format
<day>	Day of the current month <1-31>
<month>	The first three letters of the current month.
<year>	Current year <2000-2035>

Mode Privileged Exec

Usage Configure the timezone before setting the local time. Otherwise, when you change the timezone, the device applies the new offset to the local time.

Note If Network Time Protocol (NTP) is enabled, then you cannot change the time or date using this command. NTP maintains the clock automatically using an external time source. If you wish to manually alter the time or date, you must first disable NTP.

Example To set the time and date on your system to 2pm on the 2nd of April 2007, use the command:

```
awplus# clock set 14:00:00 2 apr 2007
```

Related Commands clock timezone

clock summer-time date

This command defines the start and end of summertime for a specific year only, and specifies summertime's offset value to Standard Time for that year.

The **no** variant of this command removes the device's summertime setting. This clears both specific summertime dates and recurring dates (set with the **clock summer-time recurring** command on page 10.8).

By default, the device has no summertime definitions set.

Syntax

```
clock summer-time <timezone-name> date <start-day> <start-month>
   <start-year> <start-time> <end-day> <end-month> <end-year>
   <end-time> <1-180>

no clock summer-time
```

Parameter	Description
<timezone-name>	A description of the summertime zone, up to 6 characters long.
`date`	Specifies that this is a date-based summertime setting for just the specified year.
<start-day>	Day that the summertime starts, in the range 1-31.
<start-month>	First three letters of the name of the month that the summertime starts.
<start-year>	Year that summertime starts, in the range 2000-2035.
<start-time>	Time of the day that summertime starts, in the 24-hour time format HH:MM.
<end-day>	Day that summertime ends, in the range 1-31.
<end-month>	First three letters of the name of the month that the summertime ends.
<end-year>	Year that summertime ends, in the range 2000-2035.
<end-time>	Time of the day that summertime ends, in the 24-hour time format HH:MM.
<1-180>	The offset in minutes.

Mode Global Configuration

Examples To set a summertime definition for New Zealand using NZST (UTC+12:00) as the standard time, and NZDT (UTC+13:00) as summertime, with the summertime set to begin on the 1st October 2007 and end on the 18th of March 2008:

```
awplus(config)# clock summer-time NZDT date 1 oct 2:00 2007 18
                mar 2:00 2008 60
```

To remove any summertime settings on the system, use the command:

```
awplus(config)# no clock summer-time
```

Related Commands
clock summer-time recurring
clock timezone

clock summer-time recurring

This command defines the start and end of summertime for every year, and specifies summertime's offset value to Standard Time.

The **no** variant of this command removes the device's summertime setting. This clears both specific summertime dates (set with the **clock summer-time date** command on page 10.7) and recurring dates.

By default, the device has no summertime definitions set.

Syntax

```
clock summer-time <timezone-name> recurring <start-week> <start-day>
   <start-month> <start-time> <end-week> <end-day> <end-month>
   <end-time> <1-180>

no clock summer-time
```

Parameter	Description
<timezone-name>	A description of the summertime zone, up to 6 characters long.
`recurring`	Specifies that this summertime setting applies every year from now on.
<start-week>	Week of the month when summertime starts, in the range 1-5. The value 5 indicates the last week that has the specified day in it for the specified month. For example, to start summertime on the last Sunday of the month, enter **5** for *<start-week>* and **sun** for *<start-day>*.
<start-day>	Day of the week when summertime starts. Valid values are **mon**, **tue**, **wed**, **thu**, **fri**, **sat** or **sun**.
<start-month>	First three letters of the name of the month that summertime starts.
<start-time>	Time of the day that summertime starts, in the 24-hour time format HH:MM.
<end-week>	Week of the month when summertime ends, in the range 1-5. The value 5 indicates the last week that has the specified day in it for the specified month. For example, to end summertime on the last Sunday of the month, enter **5** for *<end-week>* and **sun** for *<end-day>*.
<end-day>	Day of the week when summertime ends. Valid values are **mon**, **tue**, **wed**, **thu**, **fri**, **sat** or **sun**.
<end-month>	First three letters of the name of the month that summertime ends.
<end-time>	Time of the day that summertime ends, in the 24-hour time format HH:MM.
<1-180>	The offset in minutes.

Mode Global Configuration

Examples

To set a summertime definition for New Zealand using NZST (UTC+12:00) as the standard time, and NZDT (UTC+13:00) as summertime, with summertime set to start on the 1st Sunday in October, and end on the 3rd Sunday in March, use the command:

```
awplus(config)# clock summer-time NZDT recurring 1 sun oct 2:00
                3 sun mar 2:00 60
```

To remove any summertime settings on the system, use the command:

```
awplus(config)# no clock summer-time
```

Related Commands

clock summer-time date
clock timezone

clock timezone

This command defines the device's clock timezone. The timezone is set as a offset to the UTC.

The **no** variant of this command resets the system time to UTC.

By default, the system time is set to UTC.

Syntax

```
clock timezone <timezone-name> {minus|plus} [<0-13>|<0-12>:<00-59>]
no clock timezone
```

Parameter	Description
<timezone-name>	A description of the timezone, up to 6 characters long.
`minus` or `plus`	The direction of offset from UTC. The **minus** option indicates that the timezone is behind UTC. The **plus** option indicates that the timezone is ahead of UTC.
<0-13>	The offset in hours or from UTC.
<0-12>:<00-59>	The offset in hours or from UTC.

Mode Global Configuration

Usage Configure the timezone before setting the local time. Otherwise, when you change the timezone, the device applies the new offset to the local time.

Examples To set the timezone to New Zealand Standard Time with an offset from UTC of +12 hours, use the command:

```
awplus(config)# clock timezone NZST plus 12
```

To set the timezone to Indian Standard Time with an offset from UTC of +5:30 hours, use the command:

```
awplus(config)# clock timezone IST plus 5:30
```

To set the timezone back to UTC with no offsets, use the command:

```
awplus(config)# no clock timezone
```

Related Commands
clock set
clock summer-time date
clock summer-time recurring

continuous-reboot-prevention

Use this command to enable and to configure the continuous reboot prevention feature. Continuous reboot prevention allows the user to configure the time period during which reboot events are counted, the maximum number of times the switch can reboot within the specified time period, referred to as the threshold, and the action to take if the threshold is exceeded.

Use the **no** variant of this command to disable the continuous reboot prevention feature or to return the **period**, **threshold** and **action** parameters to the defaults.

Syntax

```
continuous-reboot-prevention enable

continuous-reboot-prevention [period <0-604800>] [threshold <1-10>]
    [action [linkdown|logonly|stopreboot]]

no continuous-reboot-prevention enable

no continuous-reboot-prevention [period] [threshold] [action]}
```

Parameter	Description	
`enable`	Enable the continuous reboot prevention feature.	
`period`	Set the period of time in which reboot events are counted.	
	<0-604800>	Period value in seconds. The default is 600.
`threshold`	Set the maximum number of reboot events allowed in the specified period.	
	<1-10>	Threshold value. The default is 1.
`action`	Set the action taken if the threshold is exceeded.	
	`linkdown`	Reboot procedure continues and all switch ports and stack ports stay link-down. The reboot event is logged. This is the default action.
	`logonly`	Reboot procedure continues normally and the reboot event is logged.
	`stopreboot`	Reboot procedure stops until the user enters the key "c" via the CLI. Normal reboot procedure then continues and the reboot event is logged.

Default Continuous reboot prevention is disabled by default. The default `period` value is `600`, the default `threshold` value is `1` and the default `action` is `linkdown`.

Mode Global Configuration

Examples To enable continuous reboot prevention, use the commands:

```
awplus# configure terminal
awplus(config)# continuous-reboot-prevention enable
```

To set the `period` to `500` and `action` to `stopreboot`, use the commands:

```
awplus# configure terminal
awplus(config)# continuous-reboot-prevention period 500
action stopreboot
```

To return the `period` and `action` to the defaults and keep the continuous reboot prevention feature enabled, use the commands:

```
awplus# configure terminal
awplus(config)# no continuous-reboot-prevention period action
```

To disable continuous reboot prevention, use the commands:

```
awplus# configure terminal
awplus(config)# no continuous-reboot-prevention enable
```

Related Commands

show continuous-reboot-prevention
show reboot history
show tech-support

ecofriendly led

Use this command to enable the eco-friendly LED (Light Emitting Diode) feature, which turns off power to the port LEDs, including the stack port status LEDs. Power to the system status, SD and stack management LEDs is not disabled.

Use the **no** variant of this command to disable the eco-friendly LED feature.

Syntax

```
ecofriendly led
no ecofriendly led
```

Default The eco-friendly LED feature is disabled by default.

Mode Global Configuration

Usage When the eco-friendly LED feature is enabled, a change in port status will not affect the display of the associated LED. When the eco-friendly LED feature is disabled and power is returned to port LEDs, the LEDs will correctly show the current state of the ports.

In a stack environment, enabling the eco-friendly LED feature on the stack master will apply the feature to every member of the stack.

For an example of how to configure a trigger to enable the eco-friendly LED feature, see **"Turn Off Power to Port LEDs" on page 102.7**. See also the section **"Save Power with the Eco-Friendly Feature" on page 1.32**.

Examples To enable the eco-friendly LED feature which turns off power to all port LEDs, use the following commands:

```
awplus# configure terminal
awplus(config)# ecofriendly led
```

To disable the eco-friendly LED feature, use the following command:

```
awplus# configure terminal
awplus(config)# no ecofriendly led
```

Related Commands

ecofriendly lpi
show ecofriendly

ecofriendly lpi

Use this command to conserve power by enabling the eco-friendly LPI (Low Power Idle) feature. This feature reduces the power supplied to the ports by the switch whenever the ports are idle and are connected to IEEE 802.3az Energy Efficient Ethernet compliant host devices. See the section **"Save Power with the Eco-Friendly Feature" on page 1.32**.

LPI is a feature of the IEEE 802.3az Energy Efficient Ethernet (EEE) standard. LPI lowers power consumption of switch ports during periods of low link utilization when connected to IEEE 802.3az compliant host devices. If no data is sent then the switch port can enter a sleep state, called Low Power Idle (LPI), to conserve power used by the switch.

Use the **no** variant of this command to disable the eco-friendly LPI feature.

Syntax

```
ecofriendly lpi
no ecofriendly lpi
```

Default The eco-friendly LPI feature is disabled by default.

Mode Interface Configuration for a switch port, or Interface Configuration for a range of switch ports.

Usage For an example of how to configure a trigger to enable the eco-friendly LPI feature, see **"Reduce Power Supplied to Ports" on page 102.9**.

All ports configured for LPI must support LPI in hardware and must be configured to auto negotiate by default or by using the **speed** and **duplex** commands as needed.

Examples To enable the eco-friendly LPI feature on a switch port, `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# ecofriendly lpi
```

To enable the eco-friendly LPI feature on a range of switch ports, `port1.0.2-port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.4
awplus(config-if)# ecofriendly lpi
```

To disable the eco-friendly feature on `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no ecofriendly lpi
```

To disable the eco-friendly feature on a range of switch ports, `port1.0.2-port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.4
awplus(config-if)# no ecofriendly lpi
```

Related Commands

duplex
ecofriendly led
show ecofriendly
show interface
speed

findme

Use this command to physically locate a specific device from a group of similar devices. Activating the command causes a selected number of port LEDs to alternately flash green then amber (if that device has amber LEDs) at a rate of 1 Hz.

Use the **no** variant of this command to deactivate the Find Me feature prior to the timeout expiring.

Syntax

```
findme [interface <port-list>|member <stack-ID>] [timeout <duration>]
no findme
```

Parameter	Description
<port-list>	The ports to flash. The port list can be: ■ a switch port, e.g. `port1.0.4` ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.4` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.5-1.0.6`.
<stack-ID>	Stack member number, from 1 to 4.
<duration>	Specify the duration in seconds within the range of 5-3600 seconds.

Default By default all port LEDs flash for 60 seconds.

Mode Privileged Exec

Usage Running the **findme** command causes the device's port LEDs to flash. An optional **timeout** parameter specifies the flash behavior duration. Normal LED behavior is restored automatically after either the default time, or a specified time has elapsed, or a **no findme** command is used. You can specify which interface or interfaces are flashed with the optional **interface** parameter.

You can specify a particular stack member with the optional **member** parameter. All available interfaces are flashed by default.

Note The **interface** and **member** parameters are mutually exclusive.

Example To activate the Find Me feature for the default duration (60 seconds) on all ports, use the following command:

```
awplus# findme
```

To activate the Find Me feature for 120 seconds on all ports, use the following command:

```
awplus# findme timeout 120
```

To activate the Find Me feature for the default duration (60 seconds) on switch port interfaces `port1.0.2` through `port1.0.4`, use the following command:

```
awplus# findme interface port1.0.2-1.0.4
```

In the example above, ports 2 to 4 will flash 4 times and then all ports will flash twice. Each alternate flash will be amber (if that device has amber LEDs). This pattern will repeat until **timeout** (default or set) or **no findme** commands are used.

To deactivate the Find Me feature, use the following command:

```
awplus# no findme
```

To activate the Find Me feature for the default duration on stack member 2, use the following command:

```
awplus# findme member 2
```

In the example above, all ports on member 2 will flash 4 times and then all ports in the stack will flash twice. Each alternate flash will be amber (if that device has amber LEDs). This pattern will repeat until the **timeout** (default or set) expires or the **no findme** commands is used.

hostname

This command sets the name applied to the device as shown at the prompt. The hostname is:

- displayed in the output of the **show system** command
- displayed in the CLI prompt so you know which device you are configuring
- stored in the MIB object sysName

On a stack, after the stack master is elected, the master will have a host name: `awplus` by default, and this also becomes the name of the stack. Individual stack members (excluding the master) will have a host name that is the stack name hyphenated with a numeric suffix. For example, `awplus-1`, `awplus-2` and so on.

The hostname command can then be used to change the stack name and the stack master's host name. For example, for the hostname `Lab` the stack master's host name will be `Lab` and the other stack members will have host names `Lab-1`, `Lab-2` and so on.

In case of stack master fail-over, or stack split, the new stack will use the previous stack name as its host name and the stack name, unless it is changed by executing the hostname command on the new stack master.

Use the **no** variant of this command to revert the hostname setting to its default (`awplus`).

Syntax `hostname <hostname>`

`no hostname [<hostname>]`

Parameter	Description
<hostname>	Specifies the name given to a specific switch. Also referred to as the Node Name in AMF output screens.

Default `awplus`

Mode Global Configuration

Usage The name must also follow the rules for ARPANET host names. The name must start with a letter, end with a letter or digit, and use only letters, digits, and hyphens. Refer to RFC 1035.

Note Within an AMF network, any switch without a hostname applied will automatically be assigned a name based on its MAC address.

To efficiently manage your network using AMF, we strongly advise that you devise a naming convention for your network switches and accordingly apply an appropriate hostname to each switch.

Example To set the system name to `HQ-Sales`, use the command:

```
awplus# configure terminal
awplus(config)# hostname HQ-Sales
```

This changes the prompt to:

```
HQ-Sales(config)#
```

To revert to the default hostname `awplus`, use the command:

```
HQ-Sales(config)#no hostname
```

This changes the prompt to:

```
awplus(config)#
```

Note When AMF is configured, running the **no hostname** command will apply a hostname that is based on the MAC address of the switch node, for example, **node_0016_76b1_7a5e**.

Related Commands **show system**

max-fib-routes

This command enables you to control the maximum number of FIB routes configured. It operates by providing parameters that enable you to configure preset maximums and warning message thresholds. The operation of these parameters is explained in the Parameter / Descriptions table shown below.

Note To set static routes, use the **max-static-routes** command on page 10.21.

Use the **no** variant of this command to set the maximum number of fib routes to the default of 4294967294 fib routes.

Syntax

```
max-fib-routes <1-4294967294> [<1-100>|warning-only]

no max-fib-routes
```

Parameter	Description
`max-fib-routes`	This is a the maximum number of routes that can be stored in the switch's Forwarding Information dataBase. In practice, other practical system limits would prevent this maximum being reached.
<1-4294967294>	The allowable configurable range for setting the maximum number of FIB-routes.
<1-100>	This parameter enables you to optionally apply a percentage value. This percentage will be based on the maximum number of FIB routes you have specified. This will cause a warning message to appear when your routes reach your specified percentage value. Routes can continue to be added until your configured maximum value is reached.
`warning-only`	This parameter enables you to optionally apply a warning message. If you set this option a warning message will appear if your maximum configured value is reached. Routes can continue to be added until your switch reaches either the maximum capacity value of 4294967294, or a practical system limit.

Default The default number of fib routes is the maximum number of fib routes (4294967294).

Mode Global Configuration

Examples To set the maximum number of dynamic routes to 2000 and warning threshold of 75%, use the following commands:

```
awplus# config terminal
awplus(config)# max-fib-routes 2000 75
```

max-static-routes

Use this command to set the maximum number of static routes, excluding FIB (Forwarding Information Base) routes. Note that FIB routes are set and reset using **max-fib-routes**.

Use the **no** variant of this command to set the maximum number of static routes to the default of 1000 static routes.

Note To set dynamic FIB routes, use the **max-fib-routes** command on page 10.20.

Syntax

```
max-static-routes <1-1000>
no max-static-routes
```

Default The default number of static routes is the maximum number of static routes (1000).

Mode Global Configuration

Example To reset the maximum number of static routes to the default maximum, use the command:

```
awplus# configure terminal
awplus(config)# no max-static-routes
```

Note Static routes are applied before adding routes to the RIB (Routing Information Base). Therefore, rejected static routes will not appear in the running config.

Related Commands **max-fib-routes**

no debug all

This command disables the debugging facility for all features on your device. This stops the device from generating any diagnostic debugging messages.

The debugging facility is disabled by default.

Syntax `no debug all [dot1x|ipv6|nsm|ospf|vrrp]`

Parameter	Description
`dot1x`	Turns off all debugging for IEEE 802.1X port-based network access-control.
`ipv6`	Turns off all debugging for IPv6 (Internet Protocol version 6).
`nsm`	Turns off all debugging for the NSM (Network Services Module).
`ospf`	Turns off all debugging for OSPF (Open Path Shortest First).
`vrrp`	Turns off all debugging for VRRP (Virtual Router Redundancy Protocol).

Mode Global Configuration and Privileged Exec

Example To disable debugging for all features, use the command:

```
awplus# no debug all
```

To disable all 802.1X debugging, use the command:

```
awplus# no debug all
```

To disable all IPv6 debugging, use the command:

```
awplus# no debug all
```

To disable all NSM debugging, use the command:

```
awplus# no debug all
```

To disable all OSPF debugging, use the command:

```
awplus# no debug all ospf
```

To disable all VRRP debugging, use the command:

```
awplus# no debug all vrrp
```

Related Commands **undebug all**

reboot

This command halts the device and performs a cold restart (also known as reload). It displays a confirmation request before restarting.

You can reboot a stand-alone device, a stack, or a specified stack member.

Syntax

```
reboot <stack-ID>
reload <stack-ID>
reboot
reload
```

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.

Mode Privileged Exec

Usage The **reboot** and **reload** commands perform the same action.

When restarting the whole stack, you can either use this reboot command to reboot all stack members immediately, or to minimize downtime, reboot the stack members in a rolling sequence by using the **reboot rolling** command on page 111.5.

Examples To restart the stand-alone device, use the command:

```
awplus# reboot
reboot system? (y/n): y
```

To restart all devices in the stack, use the command:

```
awplus# reboot
Are you sure you want to reboot the whole stack? (y/n): y
```

To restart stack member 2, use the command:

```
awplus# reboot stack-member 2
reboot stack-member 2 system? (y/n): y
```

If the specified stack member ID does not exist in the current stack, the command is rejected.

Related Commands

reboot rolling
reload rolling

reload

This command performs the same function as the **reboot** command on page 10.23.

show clock

This command displays the system's current configured local time and date. It also displays other clock related information such as timezone and summertime configuration.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show clock`

Mode User Exec and Privileged Exec

Example To display the system's current local time, use the command:

```
awplus# show clock
```

Output **Figure 10-1: Example output from the show clock command for a switch using New Zealand time**

```
Local Time: Mon,  6 Aug 2007 13:56:06 +1200
UTC Time:   Mon,  6 Aug 2007 01:56:06 +0000
Timezone: NZST
Timezone Offset: +12:00
Summer time zone: NZDT
Summer time starts: Last Sunday in September at 02:00:00
Summer time ends: First Sunday in April at 02:00:00
Summer time offset: 60 mins
Summer time recurring: Yes
```

Table 10-1: Parameters in the output of the show clock command

Parameter	Description
`Local Time`	Current local time.
`UTC Time`	Current UTC time.
`Timezone`	The current configured timezone name.
`Timezone Offset`	Number of hours offset to UTC.
`Summer time zone`	The current configured summertime zone name.
`Summer time starts`	Date and time set as the start of summer time.
`Summer time ends`	Date and time set as the end of summer time.
`Summer time offset`	Number of minutes that summer time is offset from the system's timezone.
`Summer time recurring`	Whether the device will apply the summer time settings every year or only once.

Related Commands
clock set
clock summer-time date
clock summer-time recurring
clock timezone

show continuous-reboot-prevention

This command displays the current continuous reboot prevention configuration.

Syntax `show continuous-reboot-prevention`

Mode User Exec and Privileged Exec

Examples To show the current continuous reboot prevention configuration, use the command:

```
awplus# show continuous-reboot-prevention
```

Output **Figure 10-2: Example output from the show continuous-reboot-prevention command**

```
-------------------------------------------
Continuous reboot prevention
-------------------------------------------
status=disabled
period=600
threshold=1
action=linkdown
-------------------------------------------
```

Related Commands continuous-reboot-prevention
show reboot history

show cpu

This command displays a list of running processes with their CPU utilization.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show cpu [<stack-ID>] [sort {thrds|pri|sleep|runtime}]`

Parameter	Description	
<stack-ID>	Stack member number, from 1 to 4.	
`sort`	Changes the sorting order using the following fields. If you do not specify a field, then the list is sorted by percentage CPU utilization.	
	`thrds`	Sort by the number of threads.
	`pri`	Sort by the process priority.
	`sleep`	Sort by the average time sleeping.
	`runtime`	Sort by the runtime of the process.

Mode User Exec and Privileged Exec

Usage Entering this command on the stack master will display the information for all the stack members. A stack member heading will distinguish the different information for every stack member device.

Examples To show the CPU utilization of current processes, sorting them by the number of threads the processes are using, use the command:

```
awplus# show cpu sort thrds
```

Note that in a stack environment, executing this command on the stack master will show CPU utilization for all stack members.

To show CPU utilization for a specific stack member (in this case stack member 2), use the following command:

```
awplus# show cpu 2
```

Output **Figure 10-3: Example output from the show cpu command**

```
Stack member 2:

CPU averages:
 1 second: 12%, 20 seconds: 2%, 60 seconds: 2%
System load averages:
 1 minute: 0.03, 5 minutes: 0.02, 15 minutes: 0.00
Current CPU load:
 userspace: 6%, kernel: 4%, interrupts: 1% iowaits: 0%

user processes
==============
  pid name            thrds  cpu%   pri state sleep% runtime
 1544 hostd              1   2.8    20   run      0   120
 1166 exfx              17   1.8    20 sleep      0  3846
 1198 stackd             1   0.9    20 sleep      0   459
 1284 aisexec           44   0.9    -2 sleep      0  2606
    1 init               1   0.0    20 sleep      0   120
 9772 sh                 1   0.0    20 sleep      0     0
 9773 corerotate         1   0.0    20 sleep      0     0
  853 syslog-ng          1   0.0    20 sleep      0   356
  859 klogd              1   0.0    20 sleep      0     1
  910 inetd              1   0.0    20 sleep      0     3
  920 portmap            1   0.0    20 sleep      0     0
  931 crond              1   0.0    20 sleep      0     1
 1090 openhpid          11   0.0    20 sleep      0   233
 1111 hpilogd            1   0.0    20 sleep      0     0
 1240 hsl                1   0.0    20 sleep      0    79
 1453 authd              1   0.0    20 sleep      0    85
 1497 cntrd              1   0.0    20 sleep      0     2
 1520 epsrd              1   0.0    20 sleep      0    56
 1571 imi                1   0.0    20 sleep      0   275
 1594 irdpd              1   0.0    20 sleep      0    23
 1617 lacpd              1   0.0    20 sleep      0    87
 1638 mstpd              1   0.0    20 sleep      0    75
 1662 nsm                1   0.0    20 sleep      0   163
 1685 ospfd              1   0.0    20 sleep      0    35
 1708 pdmd               1   0.0    20 sleep      0    23
 1729 pimd               1   0.0    20 sleep      0    32
 1751 ripd               1   0.0    20 sleep      0    33
 1775 ripngd             1   0.0    20 sleep      0    25
 1797 rmond              1   0.0    20 sleep      0    64
 1963 ntpd               1   0.0    20 sleep      0    15
 2102 atlgetty           1   0.0    20 sleep      0     0
 2712 rpc.statd          1   0.0    20 sleep      0     0
 2716 rpc.statd          1   0.0    20 sleep      0     0
 2722 rpc.mountd         1   0.0    20 sleep      0     0
 2821 automount          1   0.0    20 sleep      0    82
 2892 ntpd               1   0.0    20 sleep      0    17
 2912 sshd               1   0.0    20 sleep      0     0
 9774 login              1   0.0    20 sleep      0     2
12689 more               1   0.0    20 sleep      0     0

.
.
.
```

Table 10-2: Parameters in the output of the show cpu command

Parameter	Description
`Stack member`	Stack member number.
`CPU averages`	Average CPU utilization for the periods stated.
`System load averages`	The average number of processes waiting for CPU time for the periods stated.
`Current CPU load`	Current CPU utilization specified by load types.

Table 10-2: Parameters in the output of the show cpu command

Parameter	Description
`pid`	Identifier number of the process.
`name`	A shortened name for the process
`thrds`	Number of threads in the process.
`cpu%`	Percentage of CPU utilization that this process is consuming.
`pri`	Process priority state.
`state`	Process state; one of “run”, “sleep”, “zombie”, and “dead”.
`sleep%`	Percentage of time that the process is in the sleep state.
`runtime`	The time that the process has been running for, measured in jiffies. A jiffy is the duration of one tick of the system timer interrupt.

Related Commands

show memory
show memory allocations
show memory history
show memory pools
show process

show cpu history

This command prints a graph showing the historical CPU utilization.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show [<stack-ID>]cpu history`

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.

Mode User Exec and Privileged Exec

Usage This command's output displays three graphs of the percentage CPU utilization:

- per second for the last minute, then
- per minute for the last hour, then
- per 30 minutes for the last 30 hours.

If this command is entered on the stack master, it will print graphs for all the stack members. A stack member heading will be displayed to distinguish the different graphs for every stack member.

Examples To display a graph showing the historical CPU utilization of the device, use the command:

```
awplus# show cpu history
```

To display the CPU utilization history graph for stack member 2, use the command:

```
awplus# show 2 cpu history
```

where 2 is the node id of the stack member.

Output **Figure 10-4: Example output from the show cpu history command**

```
Stack member 2:

Per second CPU load history

100
 90
 80
 70
 60
 50
 40
 30
 20
 10 ************************************************************
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                 Newest
        CPU load% per second (last 60 seconds)
              * = average CPU load%


Per minute CPU load history

100         *+
 90         +
 80
 70
 60
 50
 40
 30
 20                     +                   +
 10          ************************************************
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                 Newest
        CPU load% per minute (last 60 minutes)
              * = average CPU load%, + = maximum


Per (30) minute CPU load history

100                                                           +
 90
 80
 70
 60
 50
 40
 30
 20
 10                                                           **
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                 Newest
        CPU load% per 30 minutes (last 60 values / 30 hours)
              * = average, - = minimum, + = maximum
.
.
.
```

Related Commands

show memory
show memory allocations
show memory pools
show process

show debugging

This command displays information for all debugging options.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging`

Default This command runs all the **show debugging** commands in alphabetical order.

Mode User Exec and Privileged Exec

Usage This command displays all debugging information, similar to the way the **show tech-support** command displays all show output for use by Allied Telesis authorized service personnel only.

Example To display all debugging information, use the command:

```
awplus# show debugging
```

Output **Figure 10-5: Example output from the show debugging command**

```
awplus#show debugging
AAA debugging status:
  Authentication debugging is off
  Accounting debugging is off
% DHCP Snooping service is disabled

802.1X debugging status:

EPSR debugging status:
 EPSR Info debugging is off
 EPSR Message debugging is off
 EPSR Packet debugging is off
 EPSR State debugging is off
IGMP Debugging status:
  IGMP Decoder debugging is off
  IGMP Encoder debugging is off
.
.
.
```

Related Commands

show debugging aaa
show debugging dot1x
show debugging epsr
show debugging igmp
show debugging ip dns forwarding
show debugging lacp
show debugging lldp
show debugging mstp
show debugging ospf
show debugging pim dense-mode
show debugging pim sparse-mode
show debugging power-inline
show debugging radius
show debugging rip
show debugging snmp
show debugging vrrp

show ecofriendly

This command displays the switch's eco-friendly configuration status. The **ecofriendly led** and **ecofriendly lpi** configuration status are shown in the **show ecofriendly** output.

Syntax `show ecofriendly`

Mode Privileged Exec and Global Configuration

Example To display the switch's eco-friendly configuration status, use the following command:

```
awplus# show ecofriendly
```

Output **Figure 10-6: Example output from the show ecofriendly command**

```
awplus#show ecofriendly
Front panel port LEDs          normal
Energy efficient ethernet
Port          Name             Configured  Status
port1.0.1     Port 1           lpi         lpi
port1.0.2                      lpi         lpi
port1.0.3                      lpi         lpi
port1.0.4                      off         off
port1.0.5                      lpi         off
port1.0.6     Port 6           off         off
port1.0.7                      off         -
port1.0.8                      off         -
port1.0.9                      off         -
port1.0.10                     off         -
...
```

Table 10-3: Parameters in the output of the show ecofriendly command

Parameter	Description
`normal`	The eco-friendly LED feature is disabled and port LEDs show the current state of the ports. This is the default setting.
`off`	The eco-friendly LED feature is enabled and power to the port LEDs is disabled.
`normal (configuration overridden by eco button)`	The eco-friendly LED feature has been disabled with the eco-switch button, overriding the configuration set with the **ecofriendly led** command. The port LEDs show the current state of the ports.
`off (configuration overridden by eco button)`	The eco-friendly LED feature has been enabled with the eco-switch button, overriding the configuration set with the **ecofriendly led** command. Power to the port LEDs is disabled.
`Port`	Displays the port number as assigned by the switch.
`Name`	Displays the port name if a name is configured for a port number.

Table 10-3: Parameters in the output of the show ecofriendly command

Parameter	Description
`Configured`	The eco-friendly LPI feature is configured on the port. Either LPI or off is displayed.
`Status`	The eco-friendly LPI feature is active on the port. Either LPI or off is displayed. Ports that are not running show a dash (-).

Related Commands

ecofriendly led
ecofriendly lpi

show interface memory

This command displays the shared memory used by either all interfaces, or the specified interface or interfaces. The output is useful for diagnostic purposes by Allied Telesis authorized service personnel.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show interface memory
show interface <port-list> memory
```

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.4`) a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix switch ports, static channel groups, and dynamic (LACP) channel groups in the same list

Mode User Exec and Privileged Exec

Example To display the shared memory used by all interfaces, use the command:

```
awplus# show interface memory
```

To display the shared memory used by `port1.0.1` and `port1.0.5 to port1.0.6`, use the command:

```
awplus# show interface port1.0.1,port1.0.5-1.0.6 memory
```

Output **Figure 10-7: Example output from the show interface *<port-list>* memory command**

```
awplus#show interface port1.0.1,port1.0.5-1.0.6 memory
Vlan blocking state shared memory usage
---------------------------------------------
Interface    shmid        Bytes Used     nattch     Status
port1.0.1    393228       512            1
port1.0.5    491535       512            1
port1.0.6    557073       512            1
```

Figure 10-8: Example output from the show interface memory command

```
awplus#show interface memory
Vlan blocking state shared memory usage
---------------------------------------------
Interface    shmid        Bytes Used     nattch     Status
port1.0.1    393228       512            1
port1.0.2    458766       512            1
port1.0.3    360459       512            1
port1.0.4    524304       512            1
port1.0.5    491535       512            1
port1.0.6    557073       512            1
port1.0.7    327690       512            1
port1.0.8    655380       512            1
port1.0.9    622611       512            1
.
.
port1.0.21   950301       512            1
port1.0.22   1048608      512            1
port1.0.23   1015839      512            1
port1.0.24   1081377      512            1
lo           425997       512            1
po1          1179684      512            1
po2          1212453      512            1
sa3          1245222      512            1
```

Related Commands

show interface brief
show interface status
show interface switchport

show memory

This command displays the memory used by each process that is currently running

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show memory [<stack-ID>] [sort {size|peak|stk}]`

Parameter	Description	
`<stack-ID>`	Stack member number, from 1 to 4.	
`sort`	Changes the sorting order for the list of processes. If you do not specify this, then the list is sorted by percentage memory utilization.	
	`size`	Sort by the amount of memory the process is currently using.
	`peak`	Sort by the amount of memory the process is currently using.
	`stk`	Sort by the stack size of the process.

Mode User Exec and Privileged Exec

Usage If this command is entered on the stack master, it will display corresponding memory utilization information for all the stack members. A stack member heading will display the information for every stack member device.

Example To display the memory used by the current running processes, use the command:

```
awplus# show memory
```

Output **Figure 10-9: Example output from the show memory command**

```
awplus#show memory

Stack member 1:

RAM total: 514920 kB; free: 382716; buffers: 16368 kB

user processes
==============
pid name            mem%   size   peak   data   stk
962 pss                6  33112  36260  27696   244
1   init               0    348   1092    288    84
797 syslog-ng          0    816   2152    752    84
803 klogd              0    184   1244    124    84
843 inetd              0    256   1256    136    84
.
.
.
```

Table 10-4: Parameters in the output of the show memory command

Parameter	Description
`Stack member`	Stack member number.
`RAM total`	Total amount of RAM memory free.
`free`	Available memory size.
`buffers`	Memory allocated kernel buffers.
`pid`	Identifier number for the process.
`name`	Short name used to describe the process.
`mem%`	Percentage of memory utilization the process is currently using.
`size`	Amount of memory currently used by the process.
`peak`	Greatest amount of memory ever used by the process.
`data`	Amount of memory used for data.
`stk`	The stack size.

Related Commands

show memory allocations
show memory history
show memory pools
show memory shared

show memory allocations

This command displays the memory allocations used by processes.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show memory allocations [<process>]`

Parameter	Description
<process>	Displays the memory allocation used by the specified process.

Mode User Exec and Privileged Exec

Example To display the memory allocations used by all processes on your device, use the command:

```
awplus# show memory allocations
```

Output **Figure 10-10: Example output from the show memory allocations command**

```
awplus#show memory allocations
Memory allocations for imi
----------------------------

Current 15093760 (peak 15093760)

Statically allocated memory:
 - binary/exe                 :    1675264
 - libraries                  :    8916992
 - bss/global data            :    2985984
 - stack                      :     139264

Dynamically allocated memory (heap):
 - total allocated            :    1351680
   - in use                   :    1282440
   - non-mmapped              :    1351680
 - maximum total allocated    :    1351680
 - total free space           :      69240
   - releasable               :      68968
   - space in freed fastbins  :         16

Context
           filename:line   allocated      freed
 +            lib.c:749          484
.
.
.
```

Related Commands

- show memory
- show memory history
- show memory pools
- show memory shared
- show tech-support

show memory history

This command prints a graph showing the historical memory usage.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show memory history <stack-ID>`

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.

Mode User Exec and Privileged Exec

Usage This command's output displays three graphs of the percentage memory utilization:

- per second for the last minute, then
- per minute for the last hour, then
- per 30 minutes for the last 30 hours.

If entered on the stack master, this command will display corresponding memory utilization information for all the stack members. A stack member heading will be displayed to distinguish the different lists for every stack member.

Examples To show a graph displaying the historical memory usage for either a single unstacked device, or a complete stack, use the command:

```
awplus# show memory history
```

To show a graph displaying the historical memory usage for specific stack member (stack member 2 in this example) within a stack, use the command:

```
awplus# show memory history 2
```

Output **Figure 10-11: Example output from the show memory history command**

```
STACK member 1:

Per minute memory utilization history

100
 90
 80
 70
 60
 50
 40************************************************************
 30
 20
 10
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                  Newest
        Memory utilization% per minute (last 60 minutes)
              * = average memory utilisation%.
.
.
.
----------------------------------------------------------------

STACK member 2:

Per minute memory utilization history

100
 90
 80
 70
 60
 50
 40************************************************************
 30
 20
 10
    |....|....|....|....|....|....|....|....|....|....|....|....
    Oldest                                                  Newest
        Memory utilization% per minute (last 60 minutes)
              * = average memory utilisation%.
.
.
.
```

Related Commands

show memory allocations
show memory pools
show memory shared
show tech-support

show memory pools

This command shows the memory pools used by processes.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show memory pools [<process>]`

Parameter	Description
<process>	Displays the memory pools used by the specified process.

Mode User Exec and Privileged Exec

Example To shows the memory pools used by processes, use the command:

```
awplus# show memory pools
```

Output **Figure 10-12: Example output from the show memory pools command**

```
awplus#show memory pools
Memory pools for imi
---------------------

Current 15290368 (peak 15290368)

Statically allocated memory:
 - binary/exe                 :    1675264
 - libraries                  :    8916992
 - bss/global data            :    2985984
 - stack                      :     139264

Dynamically allocated memory (heap):
 - total allocated            :    1548288
   - in use                   :    1479816
   - non-mmapped              :    1548288
 - maximum total allocated    :    1548288
 - total free space           :      68472
   - releasable               :      68200
   - space in freed fastbins  :         16
.
.
.
```

Related Commands
- show memory allocations
- show memory history
- show tech-support

show memory shared

This command displays shared memory allocation information. The output is useful for diagnostic purposes by Allied Telesis authorized service personnel.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show memory shared`

Mode User Exec and Privileged Exec

Example To display information about the shared memory allocation used on the switch, use the command:

```
awplus# show memory shared
```

Output **Figure 10-13: Example output from the show memory shared command**

```
awplus#show memory shared
Shared Memory Status
--------------------------
Segment allocated  = 39
Pages allocated    = 39
Pages resident     = 11

Shared Memory Limits
--------------------------
Maximum number of segments          = 4096
Maximum segment size (kbytes)       = 32768
Maximum total shared memory (pages) = 2097152
Minimum segment size (bytes)        = 1
```

Related Commands
show memory allocations
show memory history
show memory sort

show process

This command lists a summary of the current running processes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show process [<stack-ID>] [sort {cpu|mem}]`

Parameter	Description	
<stack-ID>	Stack member number, from 1 to 4.	
`sort`	Changes the sorting order for the list of processes.	
	`cpu`	Sorts the list by the percentage of CPU utilization.
	`mem`	Sorts the list by the percentage of memory utilization.

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display the information for all the stack members. A stack member heading will be displayed to distinguish the different information for every stack member.

Example To display a summary of the current running processes, use the command:

```
awplus# show process
```

To display a summary of the current running processes on stack member 2, use the command:

```
awplus# show process 2
```

Output **Figure 10-14: Example output from the show process command**

```
Stack member 2:

CPU load for 1 minute: 0%; 5 minutes: 3%; 15 minutes: 0%
RAM total: 514920 kB; free: 382600 kB; buffers: 16368 kB

user processes
==============
pid name          thrds  cpu%  mem%   pri  state  sleep%
962 pss              12     0     6    25  sleep       5
1   init              1     0     0    25  sleep       0
797 syslog-ng         1     0     0    16  sleep      88


kernel threads
==============
pid name          cpu%  pri  state  sleep%
71  aio/0         0     20   sleep  0
3   events/0      0     10   sleep  98
.
.
.
```

Table 10-5: Parameters in the output from the show process command

Parameter	Description
`Stack member`	Stack member number.
`CPU load`	Average CPU load for the given period.
`RAM total`	Total memory size.
`free`	Available memory.
`buffers`	Memory allocated to kernel buffers.
`pid`	Identifier for the process.
`name`	Short name to describe the process.
`thrds`	Number of threads in the process.
`cpu%`	Percentage of CPU utilization that this process is consuming.
`mem%`	Percentage of memory utilization that this process is consuming.
`pri`	Process priority.
`state`	Process state; one of “run”, “sleep”, “stop”, “zombie”, or “dead”.
`sleep%`	Percentage of time the process is in the sleep state.

Related Commands

show cpu
show cpu history

show reboot history

Use this command to display the switch's reboot history.

Syntax `show reboot history [<stack-ID>]`

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.

Mode User Exec and Privileged Exec

Example To show the reboot history of stack member 2, use the command:

```
awplus# show reboot history 2
```

Output **Figure 10-15: Example output from the show reboot history command**

```
awplus#show reboot history 2

Stack member 2:

<date>     <time>    <type>      <description>
------------------------
2014-01-10 01:42:04  Expected    User Request
2014-01-10 01:35:31  Expected    User Request
2014-01-10 01:16:25  Unexpected Rebooting due to critical process (network/nsm)
failure!
2014-01-10 01:11:04  Unexpected Rebooting due to critical process (network/nsm)
failure!
2014-01-09 20:46:40  Unexpected Rebooting due to VCS duplicate member-ID
2014-01-09 19:56:16  Expected    User Request
2010-01-09 20:36:06  Unexpected Rebooting due to VCS duplicate master (Continuous
reboot prevention)
2014-01-09 19:51:20  Expected    User Request
```

Table 10-6: Parameters in the output from the show reboot history command

Parameter	Description
`Unexpected`	Reboot is counted by the continuous reboot prevention feature if the reboot event occurs in the time period specified for continuous reboot prevention.
`Expected`	Reboot is not counted by continuous reboot prevention feature.
`Continuous reboot prevention`	A continuous reboot prevention event has occurred. The action taken is configured with the **continuous-reboot-prevention** command. The next time period during which reboot events are counted begins from this event.
`user request`	User initiated reboot via the CLI.

Related Commands **show continuous-reboot-prevention**
show tech-support

show router-id

Use this command to show the Router ID of the current system.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show router-id`

Mode User Exec and Privileged Exec

Example To display the Router ID of the current system, use the command:

```
awplus# show router-id
```

Output **Figure 10-16: Example output from the show router-id command**

```
awplus>show router-id
Router ID: 10.55.0.2 (automatic)
```

show system

This command displays general system information about the device, including the hardware installed, memory, and software versions loaded. It also displays location and contact details when these have been set.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system`

Mode User Exec and Privileged Exec

Example To display configuration information, use the command:

```
awplus# show system
```

Usage For a stacked configuration, if this command is entered on the stack master, it will display the information for all the stack members. A stack member heading will be displayed to distinguish the different information for every stack member.

Output **Figure 10-17: Example output from the show system command**

```
awplus#show system
Switch System Status                                    Mon Mar 10 08:42:16 2014

Board       ID  Bay   Board Name                          Rev   Serial number
-------------------------------------------------------------------------------
Base       369        x510-28GTX                          A-0   A24SCA01M
-------------------------------------------------------------------------------
RAM:  Total: 495792 kB Free: 384904 kB
Flash: 63.0MB Used: 50.9MB Available: 12.1MB
-------------------------------------------------------------------------------
Environment Status : Normal
Uptime             : 0 days 16:31:49
Bootloader version : 2.0.12


Current software   : x510-5.4.4-3.5.rel
Software version   : 5.4.4
Build date         : Mon Mar 03 13:42:20 NZST 2014

Current boot config: flash:/backup.cfg (file exists)

System Name
 awplus
System Contact

System Location
```

Related Commands **show system environment**

show system environment

This command displays the current environmental status of your device and any attached PSU, XEM, or other expansion option. The environmental status covers information about temperatures, fans, and voltage.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system environment`

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display the information for all the stack members. A stack member heading will be displayed to distinguish the different information for every stack member.

Example To display the system's environmental status, use the command:

```
awplus# show system environment
```

Output **Figure 10-18: Example output from the show system environment command**

```
Stack Environment Monitoring Status

Stack member 1:

Overall Status: Normal

Resource ID: 1  Name: x510-28GTX
ID  Sensor (Units)                      Reading  Low Limit High Limit Status
1   Fan: Fan 1 (Rpm)                       4344       3000          -      Ok
2   Voltage: 1.8V (Volts)                 1.804      1.617      1.978     Ok
3   Voltage: 1.0V (Volts)                 0.995      0.896      1.099     Ok
4   Voltage: 3.3V (Volts)                 3.291      2.960      3.613     Ok
5   Voltage: 5.0V (Volts)                 5.066      4.477      5.498     Ok
6   Voltage: 1.2V (Volts)                 1.187      1.072      1.318     Ok
7   Temp: CPU (Degrees C)                    50        -10         90     Ok
```

Related Commands **show system**

show system interrupts

Use this command to display the number of interrupts for each IRQ (Interrupt Request) used to interrupt input lines on a PIC (Programmable Interrupt Controller) on your switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system interrupts`

Mode User Exec and Privileged Exec

Example To display information about the number of interrupts for each IRQ in your device, use the command:

```
awplus# show system interrupts
```

Output **Figure 10-19: Example output from the show system interrupts command**

```
awplus>show system interrupts
           CPU0
  1:          2   CPM2 SIU  Level Enabled   0     i2c-mpc
  2:        145   CPM2 SIU  Level Enabled   0     spi-mpc
 77:          0   OpenPIC   Level Enabled   0     enet_tx
 78:          2   OpenPIC   Level Enabled   0     enet_rx
 82:          0   OpenPIC   Level Enabled   0     enet_error
 90:       5849   OpenPIC   Level Enabled   0     serial
 91:    2066672   OpenPIC   Level Enabled   0     i2c-mpc
 94:        147   OpenPIC   Level Enabled   0     cpm2_cascade
112:          5   OpenPIC   Edge  Enabled   0     phy_interrupt
114:     398714   OpenPIC   Level Enabled   0     mvPP
115:      26247   OpenPIC   Level Enabled   0     mvPP
119:          0   OpenPIC   Edge  Enabled   0     Power supply status
BAD:          0
```

Related Commands **show system environment**

show system mac

This command displays the physical MAC address available on a standalone switch, or a stack. This command also shows the virtual MAC address for a for a stack if the stack virtual MAC address feature is enabled with the **stack virtual-mac** or the **stack enable** command.

Syntax `show system mac`

Mode User Exec and Privileged Exec

Usage This command also displays the virtual MAC address, if the VCStack Plus virtual MAC address feature is enabled with the **stack virtual-mac** command.

For information about this VCStack Plus feature, see the section **Fixed or Virtual MAC Addressing** in the **VCStack Introduction** chapter.

Example To display the physical MAC address enter the following command:

```
awplus# show system mac
```

Output **Figure 10-20: Example output from the show system mac command**

```
awplus#show system mac
eccd.6d9d.4eed
```

Output **Figure 10-21: Example output showing how to use the stack virtual-mac command and the show system mac command**

```
awplus#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
awplus(config)#stack virtual-mac
% Please check that the new MAC 0000.cd37.0065 is unique within
the network.
% Save the config and restart the system for this change to take
effect.
Member1#copy run start
Building configuration...
[OK]
Member1#reload
reboot system? (y/n): y

        ... Rebooting at user request ...
Loading default configuration
....

awplus login: manager
Password:

awplus>show system mac
eccd.6d9d.4eed

Virtual MAC Address 0000.cd37.0065
```

Related Commands **stack virtual-mac**

show system pci device

Use this command to display the PCI devices on your switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system pci device`

Mode User Exec and Privileged Exec

Example To display information about the PCI devices on your switch, use the command:

```
awplus# show system pci device
```

Output **Example output from the show system pci device command**

```
awplus#show system pci device
00:0c.0 Class 0200: 11ab:00d1 (rev 01)
        Flags: bus master, 66Mhz, medium devsel, latency 128, IRQ 113
        Memory at 5ffff000 (32-bit, non-prefetchable) [size=4K]
        Memory at 58000000 (32-bit, non-prefetchable) [size=64M]

00:0d.0 Class 0200: 11ab:00d1 (rev 01)
        Flags: bus master, 66Mhz, medium devsel, latency 128, IRQ 116
        Memory at 57fff000 (32-bit, non-prefetchable) [size=4K]
        Memory at 50000000 (32-bit, non-prefetchable) [size=64M]
```

Related Commands **show system environment**
show system pci tree

show system pci tree

Use this command to display the PCI tree on your switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show system pci tree
```

Mode

User Exec and Privileged Exec

Example

To display information about the PCI tree on your switch, use the command:

```
awplus# show system pci tree
```

Output

Figure 10-22: Example output from the show system pci tree command

```
awplus>show system pci tree
-[00]-+-0c.0  11ab:00d1
      \-0d.0  11ab:00d1
```

Related Commands

show system environment
show system pci device

show system pluggable

This command displays **brief** pluggable transceiver information showing the pluggable type, the pluggable serial number, and the pluggable port on the switch. Different types of pluggable transceivers are supported in different models of switch. See your Allied Telesis dealer for more information about the models of pluggables that your switch supports.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system pluggable [<port-list>]`

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`.

Mode User Exec and Privileged Exec

Usage Entering this command will display the information for all pluggable transceivers in the system. In a stack, a separate heading will be displayed to distinguish each stack member's information.

Example To display brief information about pluggable transceivers installed in `port1.0.1` through `port1.0.4`, use the command:

```
awplus# show system pluggable port1.0.1-1.0.4
```

Output **Figure 10-23: Example output from the show system pluggable port1.0.1-1.0.4 command**

```
System Pluggable Information
Port   Manufacturer     Device           Serial Number    Datecode Type
-------------------------------------------------------------------------------
1.0.1  AGILENT          HFBR-5710L       0401312315461272 040131   1000BASE-SX
1.0.2  AGILENT          QBCU-5730R       AK0614GKF7       060408   1000BASE-T
1.0.3  AGILENT          HFBR-5710L       0305130112182696 030513   1000BASE-SX
1.0.4  AGILENT          HBCU-5710R       AK051300SM       050402   1000BASE-T
-------------------------------------------------------------------------------
```

Example To display information about the pluggable transceiver installed in `port1.0.1`, use the command:

```
awplus# show system pluggable port1.0.1
```

Output **Figure 10-24: Example output from the show system pluggable port1.0.1 command**

```
System Pluggable Information
Port   Manufacturer     Device           Serial Number    Datecode Type
-------------------------------------------------------------------------------
1.0.1  AGILENT          HFBR-5710L       0401312315461272 040131   1000BASE-SX
-------------------------------------------------------------------------------
```

Table 10-7: Parameters in the output from the show system pluggables command

Parameter	Description
`Stack member`	The stack member number.
`Port`	Specifies the vendor's name for the installed pluggable transceiver.
`Vendor Name`	Specifies the vendor's name for the installed pluggable transceiver.
`Device Name`	Specifies the device name for the installed pluggable transceiver.
`Device Type`	Specifies the device type for the installed pluggable transceiver.
`Serial Number`	Specifies the serial number for the installed pluggable transceiver.
`Manufacturing Datecode`	Specifies the manufacturing datecode for the installed pluggable transceiver. Checking the manufacturing datecode with the vendor may be useful when determining Laser Diode aging issues. See the **Trouble-shoot Fiber and Pluggable Issues** section in the **System Configuration and Monitoring Commands** chapter.
`SFP Laser Wavelength`	Specifies the laser wavelength of the installed pluggable transceiver.
`Datecode`	Specifies the manufacturing datecode for the installed pluggable transceiver. Checking the manufacturing datecode with the vendor may be useful when determining Laser Diode aging issues. See the **Trouble-shoot Fiber and Pluggable Issues** section in the **System Configuration and Monitoring Commands** chapter.
`Device Type`	Specifies the device type for the installed pluggable transceiver

Related Commands

show system environment
show system pluggable detail
show system pluggable diagnostics

show system pluggable detail

This command displays detailed pluggable transceiver information showing the pluggable type, the pluggable serial number, and the pluggable port on the switch. Different types of pluggable transceivers are supported in different models of switch. See your Allied Telesis dealer for more information about the models of pluggables that your switch supports.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show system pluggable [<port-list>] detail
```

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`.

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display detailed information about the pluggable transceivers for all the stack members. A stack member heading will be displayed to distinguish the different pluggable transceiver information for every stack member.

In addition to the information about pluggable transceivers displayed using the **show system pluggable** command (port, manufacturer, serial number, manufacturing datecode, and type information), the **show system pluggable detail** command displays the following information:

- **SFP Laser Wavelength**: Specifies the laser wavelength of the installed pluggable transceiver
- **Single mode Fiber**: Specifies the link length supported by the pluggable transceiver using single mode fiber
- **OM1 (62.5µm) Fiber**: Specifies the link length (in µm - micron) supported by the pluggable transceiver using 62.5 micron multi-mode fiber.
- **OM2 (50 µm) Fiber**: Specifies the link length (in µm - micron) supported by the pluggable transceiver using 50 micron multi-mode fiber.
- **Diagnostic Calibration**: Specifies whether the pluggable transceiver supports DDM or DOM Internal or External Calibration.
  - « **Internal** is displayed if the pluggable transceiver supports DDM or DOM Internal Calibration.
  - « **External** is displayed if the pluggable transceiver supports DDM or DOM External Calibration.
  - « - is displayed if SFP or SFP+ DDM Internal Calibration or External Calibration is not supported.
- **Power Monitoring**: Displays the received power measurement type, which can be either **OMA** (Optical Module Amplitude) or **Avg** (Average Power) measured in µW.

Note For parameters that are not supported or not specified, a hyphen is displayed instead.

Example To display detailed information about the pluggable transceivers installed in a particular port on the switch, use a command like:

```
awplus# show system pluggable port1.0.24 detail
```

To display detailed information about all the pluggable transceivers installed on the switch, use the command:

```
awplus# show system pluggable detail
```

Output **Figure 10-25: Example output from the show system pluggable detail command on a switch**

```
awplus#show system pluggable port1.0.24 detail
System Pluggable Information Detail

Port1.0.24
==========
Vendor Name:               AGILENT
Device Name:               HFCT-5710L
Device Type:               1000BASE-LX
Serial Number:             0402142241184360
Manufacturing Datecode:    040214
SFP Laser Wavelength:      -
Link Length Supported
 Single Mode Fiber :      10Km
 OM1 (62.5um) Fiber:      550m
 OM2 (50um) Fiber  :      550m
Diagnostic Calibration:    Internal
Power Monitoring:          Avg
FEC BER support:           -
```

Example To display detailed information about the pluggable transceivers installed on a stack, use the command:

```
awplus# show system pluggable detail
```

Output

Figure 10-26: Example output from the show system pluggable detail command on a stack

```
awplus#show system pluggable detail
System Pluggable Information Detail

Stack member 1:

Port1.0.24
==========
Vendor Name:               AGILENT
Device Name:               HFCT-5710L
Device Type:               1000BASE-LX
Serial Number:             0402142241184360
Manufacturing Datecode:    040214
SFP Laser Wavelength:      -
Link Length Supported
 Single Mode Fiber :      10Km
 OM1 (62.5um) Fiber:      550m
 OM2 (50um) Fiber  :      550m
Diagnostic Calibration:    Internal
Power Monitoring:          Avg
FEC BER support:           -

Stack member 2:

Port2.0.24
==========
Vendor Name:               FINISAR CORP.
Device Name:               FTRJ-8519-7D-CSC
Device Type:               1000BASE-SX
Serial Number:             P430KGY
Manufacturing Datecode:    030718
SFP Laser Wavelength:      850nm
Link Length Supported
  Single Mode Fiber :      -
  OM1 (62.5um) Fiber:      300m
  OM2 (50um) Fiber  :      550m
Diagnostic Calibration:    Internal
Power Monitoring:          OMA
FEC BER support:           Yes
```

Table 10-8: Parameters in the output from the show system pluggables detail command:

Parameter	Description
`Stack member`	The stack member number..
`Port`	Specifies the port the pluggable transceiver is installed in.
`Vendor Name`	Specifies the vendor's name for the installed pluggable transceiver.
`Device Name`	Specifies the device name for the installed pluggable transceiver.
`Device Type`	Specifies the device type for the installed pluggable transceiver..
`Serial Number`	Specifies the serial number for the installed pluggable transceiver.
`Manufacturing Datecode`	Specifies the manufacturing datecode for the installed pluggable transceiver. Checking the manufacturing datecode with the vendor may be useful when determining Laser Diode aging issues. See the **Trouble-shoot Fiber and Pluggable Issues** section in the **System Configuration and Monitoring Commands** chapter.

Parameter	Description
`SFP Laser Wavelength`	Specifies the laser wavelength of the installed pluggable transceiver.
`Single Mode Fiber`	Specifies the link length supported by the pluggable transceiver using single mode fiber.
`OM1 (62.5um) Fiber`	Specifies the link length (in µm - micron) supported by the pluggable transceiver using 62.5 micron multi-mode fiber.
`OM2 (50um) Fiber`	Specifies the link length (in µm - micron) supported by the pluggable transceiver using 50 micron multi-mode fiber.
`Diagnostic Calibration`	Specifies whether the pluggable transceiver supports DDM or DOM Internal or External Calibration: **Internal** is displayed if the pluggable transceiver supports DDM or DOM Internal Calibration. **External** is displayed if the pluggable transceiver supports DDM or DOM External Calibration. - is displayed if SFP or SFP+ DDM Internal Calibration or External Calibration is not supported.
`Power Monitoring`	Displays the received power measurement type, which can be either **OMA** (Optical Module Amplitude) or **Avg** (Average Power) measured in µW.

Related Commands

show system environment
show system pluggable
show system pluggable diagnostics

show system pluggable diagnostics

This command displays diagnostic information about SFP and SFP+ pluggable transceivers, which support Digital Diagnostic Monitoring (DDM).

Different types of pluggable transceivers are supported in different models of switch. See your switch's Datasheet for more information about the models of pluggables that your switch supports.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system pluggable [<port-list>] diagnostics`

Parameter	Description
<port-list>	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.2.24`.

Mode User Exec and Privileged Exec

Usage For a stacked configuration, if this command is entered on the stack master, it will display information about the pluggable transceivers for all the stack members. A stack member heading will be displayed to distinguish different pluggable transceiver information for every stack member.

Modern optical SFP and SFP+ transceivers support Digital Diagnostics Monitoring (DDM) functions.

Diagnostic monitoring features allow you to monitor real-time parameters of the pluggable transceiver, such as optical output power, optical input power, temperature, laser bias current, and transceiver supply voltage. Additionally, RX LOS (Loss of Signal) is shown when the received optical level is below a preset threshold. Monitor these parameters to check on the health of all transceivers, selected transceivers or a specific transceiver installed in a switch.

Examples To display detailed information about all pluggable transceivers installed on a standalone switch, use the command:

```
awplus# show system pluggable diagnostics
```

Output **Figure 10-27: Example output from the show system pluggable diagnostics command on a switch**

```
awplus#show system pluggable diagnostics
System Pluggable Information Diagnostics

Port1.0.21          Status              Alarms                    Warnings
                  Reading   Alarm     Max     Min   Warning     Max     Min
Temp: (Degrees C)  29.387       -  100.00  -40.00         -  85.000  -10.00
Vcc: (Volts)        3.339       -   3.465   3.135         -   3.400   3.200
Tx Bias: (mA)      10.192       -  37.020   3.260         -  34.520   5.760
Tx Power: (mW)     17.872       -  35.643   8.953         -  28.313  11.271
Rx Power: (mW)      0.006     Low  15.849   0.025       Low  12.589   0.040
Rx LOS:           Rx Down

Port1.0.22          Status              Alarms                    Warnings
                  Reading   Alarm     Max     Min   Warning     Max     Min
Temp: (Degrees C)  29.387       -  100.00  -40.00         -  85.000  -10.00
Vcc: (Volts)        3.378       -   3.630   2.970         -   3.465   3.135
Tx Bias: (mA)       2.802       -   6.000   1.000         -   5.000   1.000
Tx Power: (mW)      2.900       -  11.000   0.600         -  10.000   0.850
Rx Power: (mW)      1.739       -  18.000   0.000         -  10.000   0.200
Rx LOS:             Rx Up
```

To display detailed information about the pluggable transceiver installed in `port1.0.22` on a standalone switch, use the command:

```
awplus# show system pluggable diagnostics port1.0.22
```

Output **Figure 10-28: Example output from the show system pluggable diagnostics port1.0.22 command on a switch**

```
awplus#show system pluggable port1.0.22 diagnostics
System Pluggable Information Diagnostics

Port1.0.22          Status              Alarms                    Warnings
                  Reading   Alarm     Max     Min   Warning     Max      Min
Temp: (Degrees C)  29.387       -  100.00  -40.00         -  85.000   -10.00
Vcc: (Volts)        3.378       -   3.630   2.970         -   3.465    3.135
Tx Bias: (mA)       2.802       -   6.000   1.000         -   5.000    1.000
Tx Power: (mW)      2.900       -  11.000   0.600         -  10.000    0.850
Rx Power: (mW)      1.739       -  18.000   0.000         -  10.000    0.200
Rx LOS:             Rx Up
```

Table 10-9: Parameters in the output from the show system pluggables diagnostics command

Parameter	Description
`Temp (Degrees C)`	Shows the temperature inside the transceiver.
`Vcc (Volts)`	Shows voltage supplied to the transceiver.

Parameter	Description
`Tx Bias (mA)`	Shows current to the Laser Diode in the transceiver.
`Tx Power (mW)`	Shows the amount of light transmitted from the transceiver.
`Rx Power (mW)`	Shows the amount of light received in the transceiver.
`Rx LOS`	Shows when the received optical level falls below a preset threshold.

Related Commands

show system environment
show system pluggable
show system pluggable detail

show system serialnumber

This command shows the serial number information for the switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show system serialnumber`

Mode User Exec and Privileged Exec

Example To display the serial number information for the switch, use the command:

```
awplus# show system serialnumber
```

Output **Figure 10-29: Example output from the show system serialnumber command**

```
awplus#show system serialnumber
45AX5300X
```

show tech-support

This command generates system and debugging information for the switch and saves it to a file. You can optionally limit the command output to display only information for a given protocol or feature.

The command generates a large amount of output, which is saved to a file in compressed format. The output file name can be specified by outfile option. If the output file already exists, a new file name is generated with the current time stamp. If the output filename does not end with ".gz", then ".gz" is appended to the filename. Since output files may be too large for Flash on the switch we recommend saving files to either an SD card, USB storage device, or TFTP server whenever possible to avoid switch lockup. This method is not likely to be appropriate when running the working set option of ATMF across a range of physically separated switches.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

The output of this command may include the result of the following commands:

Syntax

```
show tech-support {all|[atmf|dhcpsn|epsr|igmp|ip|ipv6|mld|ospf|ospf6|
   pim|rip|ripng|stack|stp|system|tacacs+]|[outfile <filename>]}
```

Parameter	Description
`all`	Display full information
`atmf`	Display ATMF- specific information
`dhcpsn`	Display DHCP Snooping specific information
`epsr`	Display EPSR specific information
`igmp`	Display IGMP specific information
`ip`	Display IP specific information
`ipv6`	Display IPv6 specific information
`mld`	Display MLD specific information
`ospf`	Display OSPF related information
`ospf6`	Display OSPF6 specific information
`outfile`	Output file name
`pim`	Display PIM related information
`rip`	RIP related information
`ripng`	Display RIPNG specific information
`stack`	Display stacking device information
`stp`	Display STP specific information
`system`	Display general system information
`tacacs+`	Display TACACS+ information
|	Output modifier

Parameter	Description
>	Output redirection
>>	Output redirection (append)
<*filename*>	Specifies a name for the output file. If no name is specified, this file will be saved as: tech-support.txt.gz.

Default Captures **all** information for the switch.

By default the output is saved to the file '`tech-support.txt.gz`' in the current directory. If this file already exists in the current directory then a new file is generated with the time stamp appended to the file name, for example '`tech-support20080109.txt.gz`', so the last saved file is retained.

Usage This command is useful for collecting a large amount of information about all protocols or specific protocols on your switch so that it can then be analyzed for troubleshooting purposes. The output of this command can be provided to technical support staff when reporting a problem.

Mode Privileged Exec

Examples show tech-support

```
awplus# show tech-support
```

Note

You can manage your show output, or make it a more selective, by using a command modifier. For information on using show-command modifiers see **"Controlling "show" Command Output" on page 1.37**.

Output The output of this command may include the result of the following commands:

show arp
show arp security
show arp security interface
show arp security statistics
show boot
show counter dhcp-client
show counter dhcp-relay
show counter dhcp-server
show counter log
show counter mail
show counter ntp
show counter ping-poll
show counter snmp-server
show counter stack
show cpu
show cpu history
show diagnostic channel-group
show etherchannel
show etherchannel detail
show exception log
show interface

show interface brief
show ip dhcp snooping
show ip dhcp snooping acl
show ip dhcp snooping binding
show ip dhcp snooping interface
show ip dhcp snooping statistics
show ip igmp groups
show ip igmp interface
show ip igmp snooping mrouter vlan1 (see the **show ip igmp snooping mrouter** command)
show ip interface
show ip ospf
show ip ospf database
show ip ospf interface
show ip ospf neighbor
show ip ospf route
show ip pim sparse-mode bsr-router
show ip pim sparse-mode interface detail
show ip pim sparse-mode mroute detail
show ip pim sparse-mode neighbor
show ip pim sparse-mode nexthop
show ip pim sparse-mode rp mapping
show ip route
show ip source binding
show lacp-counter
show lacp sys-id
show license
show log
show log permanent
show memory
show memory allocations
show memory history
show memory pools
show ntp associations
show ntp status
show platform
show platform port
show power-inline
show reboot history
show running-config
show spanning-tree
show stack
show stack detail (see the **show stack** command)
show startup-config
show static-channel-group
show system
show system environment
show system pluggable
show users
show vlan brief (see the **show vlan** command)
show vrrp

speed (asyn)

This command changes the console speed from the switch. Note that a change in console speed is applied for subsequent console sessions. Exit the current session to enable the console speed change using the **clear line console** command.

Syntax `speed <console-speed-in-bps>`

Parameter	Description
<console-speed-in-bps>	Console speed Baud rate in bps (bits per second).
1200	1200 Baud
2400	2400 Baud
9600	9600 Baud
19200	19200 Baud
38400	38400 Baud
57600	57600 Baud
115200	115200 Baud

Default The default console speed baud rate is 9600 bps.

Mode Line Configuration

Usage This command is used to change the console (asyn) port speed. Set the console speed to match the transmission rate of the device connected to the console (asyn) port on your switch.

Example To set the terminal console (asyn0) port speed from the switch to 57600 bps, then exit the session, and log in again to enable the change, use the commands:

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# speed 57600
awplus(config-line)# exit
awplus(config)# exit
awplus# exit
```

The new console speed of 57600 bps is applied after exiting the session and before login.

```
awplus login:
Password:

awplus>
```

Related Commands
line
clear line console
show running-config
show startup-config
speed

system territory (deprecated)

This command has been deprecated in version 5.4.4-0.1. It now has no effect.

terminal monitor

Use this command to display debugging output on a terminal.

To display the cursor after a line of debugging output, press the Enter key.

Use the command **terminal no monitor** to stop displaying debugging output on the terminal, or use the timeout option to stop displaying debugging output on the terminal after a set time.

Syntax

```
terminal monitor [<1-60>]
terminal no monitor
```

Parameter	Description
<1-60>	Set a timeout between 1 and 60 seconds for terminal output.

Default Disabled

Mode User Exec and Privileged Exec

Examples To display debugging output on a terminal, enter the command:

```
awplus# terminal monitor
```

To specify timeout of debugging output after 60 seconds, enter the command:

```
awplus# terminal monitor 60
```

To stop displaying debugging output on the terminal, use the command:

```
awplus# terminal no monitor
```

Related Commands All debug commands

undebug all

This command applies the functionality of the **no debug all** command.

Chapter 11: Debugging and Logging

Introduction 11.2
Debugging 11.2
Logging to a Terminal 11.2
Turning Off Debugging 11.3
Logging 11.3
Log Outputs 11.4

Introduction

AlliedWare Plus™ has a comprehensive debugging and logging facility in various protocols and components. This chapter describes how to start/stop debugging and logging. For detailed descriptions of the commands used to configure logging, see **Chapter 12, Logging Commands**.

Debugging

Many protocols have debug commands. Debug commands, when used with the parameters, log protocol-specific information. For example, using the **debug mstp protocol** command, results in the device writing all debugging messages generated by the MSTP algorithm to the logging system.

On using a debug command, the protocol continues to generate output until the **no** parameter is used with the command. To specify where logging output is sent, and the level of events to log, use the **log** commands in **Chapter 12, Logging Commands**.

Logging to a Terminal

To start debugging to the terminal:

Step 1: Turn on the debug options by using the relevant debug command

Step 2: Run the terminal monitor command

```
awplus> enable
awplus# configure terminal
awplus(config)# debug <protocol> (parameter)
awplus(config)# exit
awplus# terminal monitor
```

Sample Output This is a sample output of the **debug rsvp events** command displayed on the terminal:

```
awplus#terminal monitor
Dec  2 16:41:49 localhost RSVP[6518]: RSVP: RSVP message sent to
10.10.23.60/32 via interface vlan2
Dec  2 16:41:57 localhost RSVP[6518]: RSVP: Received an RSVP message
of type RSVP Reservation from 192.168.0.60 via interface vlan2
Dec  2 16:41:57 localhost RSVP[6518]: RSVP: Received a RESV message
from 10.10.23.60/32
```

Turning Off Debugging

To turn off debugging, use the `no debug` or `undebug` command. When a protocol is specified with the `no debug` or `undebug` commands, debugging is stopped for the specified protocol. To stop all debugging, use the `all` parameter with these commands.

```
awplus(config)# no debug rstp
```

or

```
awplus#undebug all
```

Logging

Protocols generate important debugging messages by default, and send them to the logging system. Additional more detailed messages can be generated by enabling debugging (**"Debugging" on page 11.2**).

Messages can be filtered based on: the program that generated the message, the severity level of the message, the type of facility that generated the message, substrings within the message text.

The severity levels in order are:

- emergencies
- alerts
- critical
- errors
- warnings
- notifications
- informational
- debugging

The facility categories are:

- auth Security/authorization messages
- authpriv Security/authorization messages (private)
- cron Clock daemon
- daemon System daemons
- ftp FTP daemon
- kern Kernel messages
- lpr Line printer subsystem
- mail Mail system
- news Network news subsystem
- syslog Messages generated internally by syslogd
- user Random user-level messages
- uucp UUCP subsystem

Log Outputs

The following types of logging output are available:

- buffered
- permanent
- terminal
- console
- host
- email

Buffered log

The buffered log is a file stored in RAM on the device. Because it is stored in RAM its content does not survive a reboot of the device. A device can only have one instance of the buffered log. The buffered log is enabled by default and has a filter to include messages with a severity level of 'notifications' and above. The buffered log can be enabled or disabled using the commands:

```
awplus# configure terminal
awplus(config)# log buffered
awplus(config)# no log buffered
```

Additional filters can be added and removed using the commands described in **log buffered (filter)** command on page 12.9:

```
awplus(config)# log buffered {facility|level|msgtext|program}
awplus(config)# no log buffered {facility|level|msgtext|
program}
```

The following log buffered commands are available:

`show log`	Displays the entire contents of the buffered log.
`show log tail`	Displays the 10 most recent entries in the buffered log.
`show log tail <10-250>`	Displays a specified number of the most recent entries in the buffered log.
`show log config`	Displays the configuration of all log outputs.
`log buffered size`	Specify the amount of memory the buffered log may use.
`clear log`	Remove the contents of the buffered log (and permanent log if it exists).
`clear log buffered`	Remove the contents of the buffered log only.
`default log buffered`	Restore the buffered log to its default configuration.

Permanent log

The permanent log is a file stored in NVS on the device. The content of the permanent log is retained over a reboot.

A device can only have one instance of the permanent log. The permanent log is enabled by default and has a filter to include messages with a severity level of 'warning' and above.

The permanent log can be disabled using the command:

```
awplus# configure terminal
awplus(config)# no log permanent
```

Additional filters can be added and removed using the commands described in **log permanent (filter)**:

```
awplus# configure terminal
awplus(config)# log permanent {facility|level|msgtext|
program}
awplus(config)# no log permanent {facility|level|msgtext|
program}
```

Table 11-1: Permanent log commands

Command	Description
`show log permanent`	Display the entire contents of the permanent log.
`show log permanent tail`	Display the 10 most recent entries in the permanent log.
`show log permanent tail <10-250>`	Display a specified number of the most recent entries in the permanent log.
`show log config`	Display the configuration of all log outputs.
`log permanent size`	Specify the amount of memory the permanent log may use.
`clear log`	Remove the contents of the buffered log and permanent log.
`clear log permanent`	Remove the contents of the permanent log only.
`default log permanent`	Restore the permanent log to its default configuration.

Host log

A host log sends log messages to a remote syslog server. A device may have many syslog hosts configured. To configure or remove a host use the commands:

```
awplus# configure terminal
awplus(config)# log host <ip-addr>9
awplus(config)# no log host <ip-addr>9
```

where `<ip-addr>` is the IP address of the remote syslog server.

There are no default filters associated with host outputs when they are created. Filters can be added and removed with the **log host (filter)** command on page 12.24.

It is not possible to view the log messages sent to this type of output as they are not retained on the device. They must be viewed on the remote device. The other host log commands are:

`show log config`	Displays the configuration of all log outputs.
`log host time`	Adjust the time information in messages to a time zone other than the one configured on this device.
`default log host <ip-address>`	Restores the device default settings for log sent to a remote syslog server.

Email log

An email log sends log messages to an email address. A device may have many email logs configured. To configure or remove an email log use the commands:

```
awplus# configure terminal
awplus(config)# log email <email-address>
awplus(config)# no log email <email-address>
```

where `<email-address>` is the destination email address.

There are no default filters associated with email outputs when they are created. Filters can be added and removed with the commands described in **log email (filter)**:

```
awplus# configure terminal
awplus(config)# log email <email-address> {facility|level|
msgtext|program}
awplus(config)# no log email <email-address> {facility|
level|msgtext|program}
```

It is not possible to view the log messages sent to this type of output as they are not retained on the device. They must be viewed by the email recipient.

The other email log commands are:

`show log config`	Displays the configuration of all log outputs.
`log email time`	Adjust the time information in messages to a time zone other than the one configured on this device.
`default log email`	Restores the device default settings for log messages sent to an email address.

Note

An email server and "from" address must be configured on the device in order for email logs to work:

- mail from <*email-address*>
- mail smtpserver <*ip-address*>

where the <*email-address*> is the 'From:' field on the sent email, and the <*ip-address*> is the email's destination SMTP server.

Email logs are sent in batches of approximately 20 messages and have the subject line "Log messages"

Chapter 12: Logging Commands

Command List 12.2
clear exception log 12.2
clear log 12.2
clear log buffered 12.3
clear log permanent 12.3
default log buffered 12.4
default log console 12.4
default log email 12.5
default log host 12.5
default log monitor 12.6
default log permanent 12.6
exception coredump size (deleted) 12.7
log buffered 12.8
log buffered (filter) 12.9
log buffered size 12.12
log console 12.13
log console (filter) 12.14
log email 12.17
log email (filter) 12.18
log email time 12.21
log host 12.23
log host (filter) 12.24
log host time 12.27
log monitor (filter) 12.29
log permanent 12.32
log permanent (filter) 12.33
log permanent size 12.36
log-rate-limit nsm 12.37
show counter log 12.38
show exception log 12.39
show log 12.40
show log config 12.42
show log permanent 12.45
show running-config log 12.46

Command List

This chapter provides an alphabetical reference of commands used to configure logging.

clear exception log

This command resets the contents of the exception log, but does not remove the associated core files.

Note When this command is used within a stacked environment, it will remove the contents of the exception logs in all stack members.

Syntax `clear exception log`

Mode Privileged Exec

Example

```
awplus# clear exception log
```

clear log

This command removes the contents of the buffered and permanent logs.

Note When this command is used within a stacked environment, it will remove the contents of the buffered and permanent logs in all stack members.

Syntax `clear log`

Mode Privileged Exec

Example To delete the contents of the buffered and permanent log use the command:

```
awplus# clear log
```

Validation Commands show log

Related Commands clear log buffered
clear log permanent

clear log buffered

This command removes the contents of the buffered log.

Note When this command is used within a stacked environment, it will remove the contents of the buffered logs in all stack members.

Syntax `clear log buffered`

Mode Privileged Exec

Example To delete the contents of the buffered log use the following commands:

```
awplus# clear log buffered
```

Validation Commands show log

Related Commands clear log
clear log permanent

clear log permanent

This command removes the contents of the permanent log.

Note When this command is used within a stacked environment, it will remove the contents of the buffered logs in all stack members.

Syntax `clear log permanent`

Mode Privileged Exec

Example To delete the contents of the permanent log use the following commands:

```
awplus# clear log permanent
```

Validation Commands show log

Related Commands clear log
clear log buffered

default log buffered

This command restores the default settings for the buffered log stored in RAM. By default the size of the buffered log is 50 kB and it accepts messages with the severity level of "warnings" and above.

Syntax `default log buffered`

Default The buffered log is enabled by default.

Mode Global Configuration

Example To restore the buffered log to its default settings use the following commands:

```
awplus# configure terminal
awplus(config)# default log buffered
```

Validation Commands **show log config**

Related Commands **log buffered**
log buffered size

default log console

This command restores the default settings for log messages sent to the terminal when a **log console** command is issued. By default all messages are sent to the console when a **log console** command is issued.

Syntax `default log console`

Mode Global Configuration

Example To restore the log console to its default settings use the following commands:

```
awplus# configure terminal
awplus(config)# default log console
```

Validation Commands **show log config**

Related Commands **log console**
log console (filter)

default log email

This command restores the default settings for log messages sent to an email address. By default no filters are defined for email addresses. Filters must be defined before messages will be sent. This command also restores the remote syslog server time offset value to local (no offset).

Syntax `default log email <email-address>`

Parameter	Description
<email-address>	The email address to send log messages to

Mode Global Configuration

Example To restore the default settings for log messages sent to the email address `admin@alliedtelesis.com` use the following commands:

```
awplus# configure terminal
awplus(config)# default log email admin@alliedtelesis.com
```

Related Commands **show log config**

default log host

This command restores the default settings for log sent to a remote syslog server. By default no filters are defined for remote syslog servers. Filters must be defined before messages will be sent. This command also restores the remote syslog server time offset value to local (no offset).

Syntax `default log host <ip-addr>`

Parameter	Description
<ip-addr>	The IP address of a remote syslog server

Mode Global Configuration

Example To restore the default settings for messages sent to the remote syslog server with IP address `10.32.16.21` use the following commands:

```
awplus# configure terminal
awplus(config)# default log host 10.32.16.21
```

Validation Commands **show log config**

Related Commands **log email**

default log monitor

This command restores the default settings for log messages sent to the terminal when a **terminal monitor** command is used.

Syntax `default log monitor`

Default All messages are sent to the terminal when a **terminal monitor** command is used.

Mode Global Configuration

Example To restore the log monitor to its default settings use the following commands:

```
awplus# configure terminal
awplus(config)# default log monitor
```

Related Commands
log monitor (filter)
show log config

default log permanent

This command restores the default settings for the permanent log stored in NVS. By default, the size of the permanent log is 50 kB and it accepts messages with the severity level of `warnings` and above.

Syntax `default log permanent`

Default The permanent log is enabled by default.

Mode Global Configuration

Example To restore the permanent log to its default settings use the following commands:

```
awplus# configure terminal
awplus(config)# default log permanent
```

Related Commands
log permanent
log permanent size
show log config

exception coredump size (deleted)

This command has been deprecated in the 5.4.4 software version, and deleted in the 5.4.4-1.1 version. There are no alternative commands.

This command sets the size of core files, and can also be used to stop core files being created.

Use the **no** variant of this command to restore the core file size to its default (unlimited).

This setting only applies to processes created after this command has been executed, to ensure this is applied to all processes the system will need to be restarted.

Syntax

```
exception coredump size {none|small|medium|large|unlimited}
no exception coredump size
```

Parameter	Description
`none`	Don't create corefiles
`small`	Create small corefiles
`medium`	Create medium corefiles
`large`	Create large corefiles
`unlimited`	Create corefiles as large as necessary (default)

Default Unlimited

Mode Global Configuration

Usage Core files are generated when a process crashes. The size of a core file can vary, its upper limit is controlled by this command. Files larger than this limit will be truncated by reducing the amount of stack and variable information stored.

Truncated core files may make debugging the failure difficult if not impossible. Reducing the amount of data stored in a core file is not recommended, however the facility is provided to reduce the amount of flash used.

Examples To restrict the size of the core file created, use the command:

```
awplus# configure terminal
awplus(config)# exception coredump size small
```

To restore the size of the core files created to the default of `unlimited`, use the command:

```
awplus# configure terminal
awplus(config)# no exception coredump size
```

log buffered

This command configures the device to store log messages in RAM. Messages stored in RAM are not retained on the device over a restart. Once the buffered log reaches its configured maximum allowable size old messages will be deleted to make way for new ones.

Syntax

```
log buffered
no log buffered
```

Default The buffered log is configured by default.

Mode Global Configuration

Examples To configured the device to store log messages in RAM use the following commands:

```
awplus# configure terminal
awplus(config)# log buffered
```

To configure the device to not store log messages in a RAM buffer use the following commands:

```
awplus# configure terminal
awplus(config)# no log buffered
```

Validation Commands show log config

Related Commands default log buffered
log buffered (filter)
log buffered size

log buffered (filter)

Use this command to create a filter to select messages to be sent to the buffered log. Selection can be based on the priority/ severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

The **no** variant of this command removes the corresponding filter, so that the specified messages are no longer sent to the buffered log.

Syntax

```
log buffered [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]

no log buffered [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages to the buffered log by severity level.
<level>	The minimum severity of message to send to the buffered log. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0 emergencies: System is unusable 1 alerts Action must be taken immediately 2 critical Critical conditions 3 errors Error conditions 4 warnings Warning conditions 5 notices Normal, but significant, conditions 6 informational Informational messages 7 debugging Debug-level messages
`program`	Filter messages to the buffered log by program. Include messages from a specified program in the buffered log.

Parameter	Description
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case-sensitive) that you find in the log output. rip Routing Information Protocol (RIP) ripng Routing Information Protocol - next generation (RIPng) ospf Open Shortest Path First (OSPF) ospfv3 Open Shortest Path First (OSPF) version 3 (OSPFv3) rsvp Resource Reservation Protocol (RSVP) pim-dm Protocol Independent Multicast - Dense Mode (PIM-DM) pim-sm Protocol Independent Multicast - Sparse Mode (PIM-SM) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI) imish Integrated Management Interface Shell (IMISH) epsr Ethernet Protection Switched Rings (EPSR) rmon Remote Monitoring loopprot Loop Protection poe Power-inline (Power over Ethernet) dhcpsn DHCP snooping (DHCPSN)
`facility`	Filter messages to the buffered log by syslog facility.
<facility>	Specify one of the following syslog facilities to include messages from in the buffered log: kern Kernel messages user Random user-level messages mail Mail system daemon System daemons auth Security/authorization messages syslog Messages generated internally by syslogd lpr Line printer subsystem news Network news subsystem uucp UUCP subsystem cron Clock daemon authpriv Security/authorization messages (private) ftp FTP daemon
`msgtext`	Select messages containing a certain text string (maximum 128 characters).
<text-string>	A text string to match (maximum 128 characters). This is case sensitive, and must be the last text on the command line.

Default By default the buffered log has a filter to select messages whose severity level is "notices (5)" or higher. This filter may be removed using the **no** variant of this command.

Mode Global Configuration

Examples To add a filter to send all messages generated by EPSR that have a severity of `notices` or higher to the buffered log use the following commands:

```
awplus# configure terminal
awplus(config)# log buffered level notices program epsr
```

To add a filter to send all messages containing the text "`Bridging initialization`", to the buffered log use the following commands:

```
awplus# configure terminal
awplus(config)# log buffered msgtext Bridging initialization
```

To remove a filter that sends all messages generated by EPSR that have a severity of `notices` or higher to the buffered log use the following commands:

```
awplus# configure terminal
awplus(config)# no log buffered level notices program epsr
```

To remove a filter that sends all messages containing the text "`Bridging initialization`", to the buffered log use the following commands:

```
awplus# configure terminal
awplus(config)# no log buffered msgtext Bridging initialization
```

Validation Commands **show log config**

Related Commands **default log buffered**
log buffered
log buffered size

log buffered size

This command configures the amount of memory that the buffered log is permitted to use. Once this memory allocation has been filled old messages will be deleted to make room for new messages.

Syntax

```
log buffered size <50-250>
```

Parameter	Description
<50-250>	Size of the RAM log in kilobytes

Mode Global Configuration

Example To allow the buffered log to use up to 100 kB of RAM use the following commands:

```
awplus# configure terminal
awplus(config)# log buffered size 100
```

Validation Commands **show log config**

Related Commands **default log buffered**
log buffered

log console

This command configures the device to send log messages to consoles. The console log is configured by default to send messages to the devices main console port.

Use the **no** variant of this command to configure the device not to send log messages to consoles.

Syntax

```
log console
no log console
```

Mode Global Configuration

Examples To configure the device to send log messages use the following commands:

```
awplus# configure terminal
awplus(config)# log console
```

To configure the device not to send log messages in all consoles use the following commands:

```
awplus# configure terminal
awplus(config)# no log console
```

Validation Commands show log config

Related Commands log console (filter)

log console (filter)

This command creates a filter to select messages to be sent to all consoles when the log console command is given. Selection can be based on the priority/severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

Syntax

```
log console [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]

no log console [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages by severity level.
<level>	The minimum severity of message to send. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0 emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages by program. Include messages from a specified program.

Parameter	Description	
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case-sensitive) that you find in the log output.	
	rip	Routing Information Protocol (RIP)
	ripng	Routing Information Protocol - next generation (RIPng)
	ospf	Open Shortest Path First (OSPF)
	ospfv3	Open Shortest Path First (OSPF) version 3 (OSPFv3)
	rsvp	Resource Reservation Protocol (RSVP)
	pim-dm	Protocol Independent Multicast - Dense Mode (PIM-DM)
	pim-sm	Protocol Independent Multicast - Sparse Mode (PIM-SM)
	dot1x	IEEE 802.1X Port-Based Access Control
	lacp	Link Aggregation Control Protocol (LACP)
	stp	Spanning Tree Protocol (STP)
	rstp	Rapid Spanning Tree Protocol (RSTP)
	mstp	Multiple Spanning Tree Protocol (MSTP)
	imi	Integrated Management Interface (IMI)
	imish	Integrated Management Interface Shell (IMISH)
	epsr	Ethernet Protection Switched Rings (EPSR)
	rmon	Remote Monitoring
	loopprot	Loop Protection
	poe	Power-inline (Power over Ethernet)
	dhcpsn	DHCP snooping (DHCPSN)
`facility`	Filter messages to the buffered log by syslog facility.	
<facility>	Specify one of the following syslog facilities to include messages from:	
	kern	Kernel messages
	user	Random user-level messages
	mail	Mail system
	daemon	System daemons
	auth	Security/authorization messages
	syslog	Messages generated internally by syslogd
	lpr	Line printer subsystem
	news	Network news subsystem
	uucp	UUCP subsystem
	cron	Clock daemon
	authpriv	Security/authorization messages (private)
	ftp	FTP daemon
`msgtext`	Select messages containing a certain text string	
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.	

Default By default the buffered log has a filter to select messages whose severity level is `critical` or higher. This filter may be removed using the **no** variant of this command. This filter may be removed and replaced by filters that are more selective.

Mode Global Configuration

Examples To create a filter to send all messages generated by MSTP that have a severity of `info` or higher to console instances where the log console command has been given, remove the default filter that includes everything use the following commands:

```
awplus# configure terminal
awplus(config)# log console level info program mstp
```

and then use the command:

```
awplus(config)# log console level info program mstp
```

To create a filter to send all messages containing the text "`Bridging initialization`" to console instances where the log console command has been given use the following commands:

```
awplus# configure terminal
awplus(config)# log console msgtext "Bridging initialization"
```

To remove a filter that sends all messages generated by EPSR that have a severity of `notices` or higher to consoles use the following commands:

```
awplus# configure terminal
awplus(config)# no log console level notices program epsr
```

To remove a default filter that includes sending `critical`, `alert` and `emergency` level messages to the console use the following commands:

```
awplus# configure terminal
awplus(config)# no log console level critical
```

Validation Commands **show log config**

Related Commands **log console**

log email

This command configures the device to send log messages to an email address. The email address is specified in this command.

Syntax `log email <email-address>`

Parameter	Description
<email-address>	The email address to send log messages to

Default By default no filters are defined for email log targets. Filters must be defined before messages will be sent.

Mode Global Configuration

Example To have log messages emailed to the email address `admin@alliedtelesis.com` use the following commands:

```
awplus# configure terminal
awplus(config)# log email admin@alliedtelesis.com
```

Validation Commands **show log config**

Related Commands **default log email**
log email

log email (filter)

This command creates a filter to select messages to be sent to an email address. Selection can be based on the priority/ severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

The **no** variant of this command configures the device to no longer send log messages to a specified email address. All configuration relating to this log target will be removed.

Syntax

```
log email <email-address> [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]

no log email <email-address> [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
<email-address>	The email address to send logging messages to
`level`	Filter messages by severity level.
<level>	The minimum severity of messages to send. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0 emergencies: System is unusable 1 alerts Action must be taken immediately 2 critical Critical conditions 3 errors Error conditions 4 warnings Warning conditions 5 notices Normal, but significant, conditions 6 informational Informational messages 7 debugging Debug-level messages
`program`	Filter messages by program. Include messages from a specified program in the log.

Parameter	Description
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rip — Routing Information Protocol (RIP) ripng — Routing Information Protocol - next generation (RIPng) ospf — Open Shortest Path First (OSPF) ospfv3 — Open Shortest Path First (OSPF) version 3 (OSPFv3) rsvp — Resource Reservation Protocol (RSVP) pim-dm — Protocol Independent Multicast - Dense Mode (PIM-DM) pim-sm — Protocol Independent Multicast - Sparse Mode (PIM-SM) dot1x — IEEE 802.1X Port-Based Access Control lacp — Link Aggregation Control Protocol (LACP) stp — Spanning Tree Protocol (STP) rstp — Rapid Spanning Tree Protocol (RSTP) mstp — Multiple Spanning Tree Protocol (MSTP) imi — Integrated Management Interface (IMI) imish — Integrated Management Interface Shell (IMISH) epsr — Ethernet Protection Switched Rings (EPSR) rmon — Remote Monitoring loopprot — Loop Protection poe — Power-inline (Power over Ethernet) dhcpsn — DHCP snooping (DHCPSN)
`facility`	Filter messages to the log by syslog facility.
<facility>	Specify one of the following syslog facilities to include messages from in the log: kern — Kernel messages user — Random user-level messages mail — Mail system daemon — System daemons auth — Security/authorization messages syslog — Messages generated internally by syslogd lpr — Line printer subsystem news — Network news subsystem uucp — UUCP subsystem cron — Clock daemon authpriv — Security/authorization messages (private) ftp — FTP daemon
`msgtext`	Select messages containing a certain text string
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.

Mode Global Configuration

Examples To create a filter to send all messages generated by EPSR that have a severity of `notices` or higher to the email address `admin@homebase.com` use the following commands:

```
awplus# configure terminal
awplus(config)# log email admin@homebase.com level notices
program epsr
```

To create a filter to send all messages containing the text "`Bridging initialization`", to the email address `admin@homebase.com` use the following commands:

```
awplus# configure terminal
awplus(config)# log email admin@homebase.com msgtext
"Bridging initialization"
```

To create a filter to send messages with a severity level of `informational` and above to the email address `admin@alliedtelesis.com` use the following commands:

```
awplus# configure terminal
awplus(config)# log email admin@alliedtelesis.com level
informational
```

To stop the device emailing log messages emailed to the email address `admin@alliedtelesis.com` use the following commands:

```
awplus# configure terminal
awplus(config)# no log email admin@homebase.com
```

To remove a filter that sends all messages generated by EPSR that have a severity of `notices` or higher to the email address `admin@homebase.com` use the following commands:

```
awplus# configure terminal
awplus(config)# no log email admin@homebase.com level
notices program epsr
```

To remove a filter that sends messages with a severity level of `informational` and above to the email address `admin@alliedtelesis.com` use the following commands:

```
awplus# configure terminal
awplus(config)# no log email admin@alliedtelesis.com level
informational
```

Related Commands **default log email**
log email
show log config

log email time

This command configures the time used in messages sent to an email address. If the syslog server is in a different time zone to your switch then the time offset can be configured using either the **utc-offset** parameter option keyword or the **local-offset** parameter option keyword, where **utc-offset** is the time difference from UTC (Universal Time, Coordinated) and **local-offset** is the difference from local time.

Syntax

```
log email <email-address> time {local|local-offset|utc-offset {plus|
   minus}<0-24>}
```

Parameter	Description
<email-address>	The email address to send log messages to
`time`	Specify the time difference between the email recipient and the switch you are configuring.
`local`	The switch is in the same time zone as the email recipient
`local-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from local time of the switch to the email recipient in hours.
`utc-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from UTC time of the switch to the email recipient in hours.
`plus`	Negative offset (difference) from the switch to the email recipient.
`minus`	Positive offset (difference) from the switch to the email recipient.
<0-24>	World Time zone offset in hours

Default The default is **local** time.

Mode Global Configuration

Usage Use the **local** option if the email recipient is in the same time zone as this device. Messages will display the time as on the local device when the message was generated.

Use the **offset** option if the email recipient is in a different time zone to this device. Specify the time offset of the email recipient in hours. Messages will display the time they were generated on this device but converted to the time zone of the email recipient.

Examples To send messages to the email address `test@home.com` in the same time zone as the switch's local time zone, use the following commands:

```
awplus# configure terminal
awplus(config)# log email admin@base.com time local 0
```

To send messages to the email address `admin@base.com` with the time information converted to the time zone of the email recipient, which is 3 hours ahead of the switch's local time zone, use the following commands:

```
awplus# configure terminal

awplus(config)# log email admin@base.com time local-offset
plus 3
```

To send messages to the email address `user@remote.com` with the time information converted to the time zone of the email recipient, which is 3 hours behind the switch's UTC time zone, use the following commands:

```
awplus# configure terminal

awplus(config)# log email user@remote.com time utc-offset
minus 3
```

Validation Commands **show log config**

Related Commands **default log buffered**

log host

This command configures the device to send log messages to a remote syslog server via UDP port 514. The IP address of the remote server must be specified. By default no filters are defined for remote syslog servers. Filters must be defined before messages will be sent.

Syntax

```
log host <ip-addr>
no log host <ip-addr>
```

Parameter	Description
<ip-addr>	The IP address of a remote syslog server in dotted decimal format A.B.C.D

Mode Global Configuration

Examples To configure the device to send log messages to a remote syslog server with IP address `10.32.16.99` use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.99
```

To stop the device from sending log messages to the remote syslog server with IP address `10.32.16.99` use the following commands:

```
awplus# configure terminal
awplus(config)# no log host 10.32.16.99
```

Validation Commands **show log config**

Related Commands **default log host**

log host (filter)

This command creates a filter to select messages to be sent to a remote syslog server. Selection can be based on the priority/severity of the message, the program that generated the message, the logging facility used, a substring within the message or a combination of some or all of these.

The **no** variant of this command configures the device to no longer send log messages to a remote syslog server. The IP address of the syslog server must be specified. All configuration relating to this log target will be removed.

Syntax

```
log host <ip-addr> [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]

no log host <ip-addr> [level <level>] [program <program-name>]
    [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
<ip-addr>	The IP address of a remote syslog server
`level`	Filter messages by severity level.
<level>	The minimum severity of messages to send. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0|emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages by program. Include messages from a specified program in the log.
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rip Routing Information Protocol (RIP) ripng Routing Information Protocol - next generation (RIPng) ospf Open Shortest Path First (OSPF) ospfv3 Open Shortest Path First (OSPF) version 3 (OSPFv3) rsvp Resource Reservation Protocol (RSVP) pim-dm Protocol Independent Multicast - Dense Mode (PIM-DM) pim-sm Protocol Independent Multicast - Sparse Mode (PIM-SM) dot1x IEEE 802.1X Port-Based Access Control

Parameter	Description	
<program-name> (cont.)	lacp	Link Aggregation Control Protocol (LACP)
	stp	Spanning Tree Protocol (STP)
	rstp	Rapid Spanning Tree Protocol (RSTP)
	mstp	Multiple Spanning Tree Protocol (MSTP)
	imi	Integrated Management Interface (IMI)
	imish	Integrated Management Interface Shell (IMISH)
	epsr	Ethernet Protection Switched Rings (EPSR)
	rmon	Remote Monitoring
	loopprot	Loop Protection
	poe	Power-inline (Power over Ethernet)
	dhcpsn	DHCP snooping (DHCPSN)
`facility`	Filter messages to the log by syslog facility.	
<facility>	Specify one of the following syslog facilities to include messages from in the log:	
	kern	Kernel messages
	user	Random user-level messages
	mail	Mail system
	daemon	System daemons
	auth	Security/authorization messages
	syslog	Messages generated internally by syslogd
	lpr	Line printer subsystem
	news	Network news subsystem
	uucp	UUCP subsystem
	cron	Clock daemon
	authpriv	Security/authorization messages (private)
	ftp	FTP daemon
`msgtext`	Select messages containing a certain text string	
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.	

Mode Global Configuration

Examples To create a filter to send all messages generated by EPSR that have a severity of `notices` or higher to a remote syslog server with IP address `10.32.16.21` use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.21 level notices program epsr
```

To create a filter to send all messages containing the text "`Bridging initialization`", to a remote syslog server with IP address `10.32.16.21` use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.21 msgtext "Bridging
initialization"
```

To create a filter to send messages with a severity level of `informational` and above to the syslog server with IP address `10.32.16.21` use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.21 level informational
```

To remove a filter that sends all messages generated by EPSR that have a severity of `notices` or higher to a remote syslog server with IP address `10.32.16.21` use the following commands:

```
awplus# configure terminal
awplus(config)# no log host 10.32.16.21 level notices program
epsr
```

To remove a filter that sends all messages containing the text "`Bridging initialization`", to a remote syslog server with IP address `10.32.16.21` use the following commands:

```
awplus# configure terminal
awplus(config)# no log host 10.32.16.21 msgtext "Bridging
initialization"
```

To remove a filter that sends messages with a severity level of `informational` and above to the syslog server with IP address `10.32.16.21` use the following commands:

```
awplusawpluls# configure terminal
awplus(config)# no log host 10.32.16.21 level informational
```

Related Commands

default log host
show log config

log host time

This command configures the time used in messages sent to a remote syslog server. If the syslog server is in a different time zone to your switch then the time offset can be configured using either the **utc-offset** parameter option keyword or the **local-offset** parameter option keyword, where **utc-offset** is the time difference from UTC (Universal Time, Coordinated) and **local-offset** is the difference from local time.

Syntax

```
log host <email-address> time {local|local-offset|utc-offset {plus|
   minus} <0-24>}
```

Parameter	Description
<email-address>	The email address to send log messages to
`time`	Specify the time difference between the email recipient and the switch you are configuring.
`local`	The switch is in the same time zone as the email recipient
`local-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from local time of the switch to the email recipient in hours.
`utc-offset`	The switch is in a different time zone to the email recipient. Use the **plus** or **minus** keywords and specify the difference (offset) from UTC time of the switch to the email recipient in hours.
`plus`	Negative offset (difference) from the switch to the syslog server.
`minus`	Positive offset (difference) from the switch to the syslog server.
<0-24>	World Time zone offset in hours

Default The default is **local** time.

Mode Global Configuration

Usage Use the **local** option if the remote syslog server is in the same time zone as the switch. Messages will display the time as on the local device when the message was generated.

Use the **offset** option if the email recipient is in a different time zone to this device. Specify the time offset of the remote syslog server in hours. Messages will display the time they were generated on this device but converted to the time zone of the remote syslog server.

Examples To send messages to the remote syslog server with the IP address `10.32.16.21` in the same time zone as the switch's local time zone, use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.21 time local 0
```

To send messages to the remote syslog server with the IP address `10.32.16.12` with the time information converted to the time zone of the remote syslog server, which is 3 hours ahead of the switch's local time zone, use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.12 time local-offset plus 3
```

To send messages to the remote syslog server with the IP address `10.32.16.02` with the time information converted to the time zone of the email recipient, which is 3 hours behind the switch's UTC time zone, use the following commands:

```
awplus# configure terminal
awplus(config)# log host 10.32.16.02 time utc-offset minus 3
```

Validation Commands **show log config**

Related Commands **default log buffered**

log monitor (filter)

This command creates a filter to select messages to be sent to the terminal when the terminal monitor command is given. Selection can be based on the priority/severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

Syntax

```
log monitor [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]

no log monitor [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages to the permanent log by severity level.
<level>	The minimum severity of message to send to the log. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0|emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages to the permanent log by program. Include messages from a specified program in the log.
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output. rip Routing Information Protocol (RIP) ripng Routing Information Protocol - next generation (RIPng) ospf Open Shortest Path First (OSPF) ospfv3 Open Shortest Path First (OSPF) version 3 (OSPFv3) rsvp Resource Reservation Protocol (RSVP) pim-dm Protocol Independent Multicast - Dense Mode (PIM-DM) pim-sm Protocol Independent Multicast - Sparse Mode (PIM-SM) dot1x IEEE 802.1X Port-Based Access Control lacp Link Aggregation Control Protocol (LACP) stp Spanning Tree Protocol (STP) rstp Rapid Spanning Tree Protocol (RSTP) mstp Multiple Spanning Tree Protocol (MSTP) imi Integrated Management Interface (IMI)

Parameter	Description	
<program-name> (cont.)	imish	Integrated Management Interface Shell (IMISH)
	epsr	Ethernet Protection Switched Rings (EPSR)
	rmon	Remote Monitoring
	loopprot	Loop Protection
	poe	Power-inline (Power over Ethernet)
	dhcpsn	DHCP snooping (DHCPSN)
facility	Filter messages to the permanent log by syslog facility.	
<facility>	Specify one of the following syslog facilities to include messages from in the log:	
	kern	Kernel messages
	user	Random user-level messages
	mail	Mail system
	daemon	System daemons
	auth	Security/authorization messages
	syslog	Messages generated internally by syslogd
	lpr	Line printer subsystem
	news	Network news subsystem
	uucp	UUCP subsystem
	cron	Clock daemon
	authpriv	Security/authorization messages (private)
	ftp	FTP daemon
msgtext	Select messages containing a certain text string	
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.	

Default By default there is a filter to select all messages. This filter may be removed and replaced by filters that are more selective.

Mode Global Configuration

Examples To create a filter to send all messages generated by MSTP that have a severity of `info` or higher to terminal instances where the terminal monitor command has been given use the following commands:

```
awplus# configure terminal
awplus(config)# log monitor level info program mstp
```

To remove a filter that sends all messages generated by EPSR that have a severity of `notices` or higher to the terminal use the following commands:

```
awplus# configure terminal
awplus(config)# no log monitor level notices program epsr
```

To remove a default filter that includes sending everything to the terminal use the following commands:

```
awplus# configure terminal
awplus(config)# no log monitor level debugging
```

Validation Commands **show log config**

Related Commands **terminal monitor**

log permanent

This command configures the device to send permanent log messages to non-volatile storage (NVS) on the device. The content of the permanent log is retained over a reboot. Once the permanent log reaches its configured maximum allowable size old messages will be deleted to make way for new messages.

The **no** variant of this command configures the device not to send any messages to the permanent log. Log messages will not be retained over a restart.

Syntax

```
log permanent
no log permanent
```

Mode Global Configuration

Examples To enable permanent logging use the following commands:

```
awplus# configure terminal
awplus(config)# log permanent
```

To disable permanent logging use the following commands:

```
awplus# configure terminal
awplus(config)# no log permanent
```

Validation Commands **show log config**

Related Commands **default log permanent**
log permanent (filter)
log permanent size
show log permanent

log permanent (filter)

This command creates a filter to select messages to be sent to the permanent log. Selection can be based on the priority/ severity of the message, the program that generated the message, the logging facility used, a sub-string within the message or a combination of some or all of these.

The **no** variant of this command removes the corresponding filter, so that the specified messages are no longer sent to the permanent log.

Syntax

```
log permanent [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]

no log permanent [level <level>] [program <program-name>]
   [facility <facility>] [msgtext <text-string>]
```

Parameter	Description
`level`	Filter messages to the permanent log by severity level.
`<level>`	The minimum severity of message to send to the log. The level can be specified as one of the following numbers or level names, where 0 is the highest severity and 7 is the lowest severity: 0|emergencies: System is unusable 1|alerts Action must be taken immediately 2|critical Critical conditions 3|errors Error conditions 4|warnings Warning conditions 5|notices Normal, but significant, conditions 6|informational Informational messages 7|debugging Debug-level messages
`program`	Filter messages to the permanent log by program. Include messages from a specified program in the log.

Parameter	Description	
<program-name>	The name of a program to log messages from, either one of the following predefined program names (not case-sensitive), or another program name (case -sensitive) that you find in the log output.	
	rip	Routing Information Protocol (RIP)
	ripng	Routing Information Protocol - next generation (RIPng)
	ospf	Open Shortest Path First (OSPF)
	ospfv3	Open Shortest Path First (OSPF) version 3 (OSPFv3)
	rsvp	Resource Reservation Protocol (RSVP)
	pim-dm	Protocol Independent Multicast - Dense Mode (PIM-DM)
	pim-sm	Protocol Independent Multicast - Sparse Mode (PIM-SM)
	dot1x	IEEE 802.1X Port-Based Access Control
	lacp	Link Aggregation Control Protocol (LACP)
	stp	Spanning Tree Protocol (STP)
	rstp	Rapid Spanning Tree Protocol (RSTP)
	mstp	Multiple Spanning Tree Protocol (MSTP)
	imi	Integrated Management Interface (IMI)
	imish	Integrated Management Interface Shell (IMISH)
	epsr	Ethernet Protection Switched Rings (EPSR)
	rmon	Remote Monitoring
	loopprot	Loop Protection
	poe	Power-inline (Power over Ethernet)
	dhcpsn	DHCP snooping (DHCPSN)
`facility`	Filter messages to the permanent log by syslog facility.	
<facility>	Specify one of the following syslog facilities to include messages from in the log:	
	kern	Kernel messages
	user	Random user-level messages
	mail	Mail system
	daemon	System daemons
	auth	Security/authorization messages
	syslog	Messages generated internally by syslogd
	lpr	Line printer subsystem
	news	Network news subsystem
	uucp	UUCP subsystem
	cron	Clock daemon
	authpriv	Security/authorization messages (private)
	ftp	FTP daemon
`msgtext`	Select messages containing a certain text string	
<text-string>	A text string to match. This is case sensitive, and must be the last text on the command line.	

Default By default the buffered log has a filter to select messages whose severity level is `notices (5)` or higher. This filter may be removed using the **no** variant of this command.

Mode Global Configuration

Examples To create a filter to send all messages generated by EPSR that have a severity of `notices` or higher to the permanent log use the following commands:

```
awplus# configure terminal
awplus(config)# log permanent level notices program epsr
```

To create a filter to send all messages containing the text "`Bridging initialization`", to the permanent log use the following commands:

```
awplus# configure terminal
awplus(config)# log permanent msgtext Bridging initialization
```

Validation Commands **show log config**

Related Commands **default log permanent**
log permanent
log permanent size
show log permanent

log permanent size

This command configures the amount of memory that the permanent log is permitted to use. Once this memory allocation has been filled old messages will be deleted to make room for new messages.

Syntax `log permanent size <50-250>`

Parameter	Description
<50-250>	Size of the permanent log in kilobytes

Mode Global Configuration

Example To allow the permanent log to use up to 100 kB of NVS use the following commands:

```
awplus# configure terminal
awplus(config)# log permanent size 100
```

Validation Commands **show log config**

Related Commands **default log permanent**
log permanent

log-rate-limit nsm

This command limits the number of log messages generated by the switch for a given interval.

Use the **no** variant of this command to revert to the default number of log messages generated by the switch of up to 200 log messages per second.

Syntax

```
log-rate-limit nsm messages <message-limit> interval <time-interval>
no log-rate-limit nsm
```

Parameter	Description
<message-limit>	<1-65535> The number of log messages generated by the switch.
<time-interval>	<0-65535> The time period for log message generation in 1/100 seconds. If an interval of 0 is specified then no log message rate limiting is applied.

Default By default, the switch will allow 200 log messages to be generated per second.

Mode Global Configuration

Usage Previously, if the switch received a continuous stream of IGMP packets with errors, such as when a packet storm occurs because of a network loop, then the switch generates a lot of log messages using more and more memory, which may ultimately cause the switch to shutdown. This log rate limiting feature constrains the rate that log messages are generated by the switch.

Note that if within the given time interval, the number of log messages exceeds the limit, then any excess log messages are discarded. At the end of the time interval, a single log message is generated indicating that log messages were discarded due to the log rate limit being exceeded.

Thus if the expectation is that there will be a lot of discarded log messages due to log rate limiting, then it is advisable to set the time interval to no less than 100, which means that there would only be one log message, indicating log excessive log messages have been discarded.

Examples To limit the switch to generate up to 300 log messages per second, use the following commands:

```
awplus# configure terminal
awplus(config)# log-rate-limit nsm messages 300 interval 100
```

To return the switch the default setting, to generate up to 200 log messages per second, use the following commands:

```
awplus# configure terminal
awplus(config)# no log-rate-limit nsm
```

show counter log

This command displays log counter information.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show counter log`

Mode User Exec and Privileged Exec

Example To display the log counter information, use the command:

```
awplus# show counter log
```

Output **Figure 12-1: Example output from the show counter log command**

```
Log counters
Total Received          ......... 2328
Total Received P0       ......... 0
Total Received P1       ......... 0
Total Received P2       ......... 1
Total Received P3       ......... 9
Total Received P4       ......... 32
Total Received P5       ......... 312
Total Received P6       ......... 1602
Total Received P7       ......... 372
```

Table 12-1: Parameters in output of the show counter log command

Parameter	Description
`Total Received`	Total number of messages received by the log
`Total Received P0`	Total number of Priority 0 (Emergency) messages received
`Total Received P1`	Total number of Priority 1 (Alert) messages received
`Total Received P2`	Total number of Priority 2 (Critical) messages received
`Total Received P3`	Total number of Priority 3 (Error) messages received
`Total Received P4`	Total number of Priority 4 (Warning) messages received
`Total Received P5`	Total number of Priority 5 (Notice) messages received
`Total Received P6`	Total number of Priority 6 (Info) messages received
`Total Received P7`	Total number of Priority 7 (Debug) messages received

Related Commands **show log config**

show exception log

This command displays the contents of the exception log. When used within a stacked environment, this command will display the contents of the exception log for all the stack members.

Syntax `show exception log`

Mode User Exec and Privileged Exec

Example To display the exception log, use the command:

```
awplus# show exception log
```

Output **Figure 12-2: Example output from the show exception log command on a switch**

```
awplus#show exception log

Stack member 1:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
------------------------------------------------------------------------
2014 Jan 27 09:57:47 local7.debug awplus corehandler : Process imish (PID:3746)
signal 11, core dumped to /flash/imish-x610-5.4.3-3.7-1-1390816667-3746.tgz
2014 Jan 27 09:57:47 local7.debug awplus corehandler : Process imish (PID:2504)
signal 11, core dumped to /flash/imish-x610-5.4.3-3.7-1-1390816667-2504.tgz
2014 Jan 27 09:58:02 local7.debug awplus corehandler : Process ospfd (PID:1512)
signal 5, core dumped to /flash/ospfd-x610-5.4.3-3.7-1-1390816682-1512.tgz
------------------------------------------------------------------------
Stack member 2:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
------------------------------------------------------------------------
2014 Jan 27 09:58:16 local7.debug awplus-2 corehandler : Process imi (PID:1427)
signal 5, core dumped to /flash/imi-x610-5.4.3-3.7-2-1390816696-1427.tgz
------------------------------------------------------------------------
```

show log

This command displays the contents of the buffered log.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show log [tail [<10-250>]]`

Parameter	Description
`tail`	Display only the latest log entries.
<10-250>	Specify the number of log entries to display.

Default By default the entire contents of the buffered log is displayed.

Mode User Exec, Privileged Exec and Global Configuration

Usage If the optional **tail** parameter is specified only the latest 10 messages in the buffered log are displayed. A numerical value can be specified after the **tail** parameter to select how many of the latest messages should be displayed.

Examples To display the contents of the buffered log use the command:

```
awplus# show log
```

To display the 10 latest entries in the buffered log use the command:

```
awplus# show log tail 10
```

Output **Figure 12-3: Example output from the show log command**

```
awplus#show log

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
------------------------------------------------------------------
2011 Aug 29 07:55:22 kern.notice awplus kernel: Linux version 2.6.32.12-at1 (mak
er@awpmaker03-dl) (gcc version 4.3.3 (Gentoo 4.3.3-r3 p1.2, pie-10.1.5) ) #1 Wed
 Dec 8 11:53:40 NZDT 2010
2011 Aug 29 07:55:22 kern.warning awplus kernel: No pci config register base in
dev tree, using default
2011 Aug 29 07:55:23 kern.notice awplus kernel: Kernel command line: console=tty
S0,9600 releasefile=x510-5.4.4-3.5.rel ramdisk=14688 bootversion=1.1.0-rc12
loglevel=1
 extraflash=00000000
2011 Aug 29 07:55:25 kern.notice awplus kernel: RAMDISK: squashfs filesystem fou
nd at block 0
2011 Aug 29 07:55:28 kern.warning awplus kernel: ipifwd: module license 'Proprie
tary' taints kernel.
.
.
.
```

Figure 12-4: Example output from the show log tail command

```
awplus#show log tail

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
------------------------------------------------------------------
2006 Nov 10 13:30:01 cron.notice crond[116]: USER manager pid 469 cmd logrotate /
etc/logrotate.conf
2006 Nov 10 13:30:01 cron.notice crond[116]: USER manager pid 471 cmd nbqueue --
wipe
2006 Nov 10 13:35:01 cron.notice crond[116]: USER manager pid 472 cmd nbqueue --
wipe
2006 Nov 10 13:40:01 cron.notice crond[116]: USER manager pid 477 cmd nbqueue --
wipe
2006 Nov 10 13:44:36 syslog.notice syslog-ng[67]: Log statistics;
processed=\'center(queued)=70\', processed=\'2006 Nov 10 13:45:01 cron.notice
crond[116]: USER manager pid 478 cmd logrotate /etc/logrotate.conf
2006 Nov 10 13:45:01 cron.notice crond[116]: USER manager pid 480 cmd nbqueue --
wipe
2006 Nov 10 13:49:32 syslog.notice syslog-ng[67]: SIGHUP received, reloading
configuration;
2006 Nov 10 13:50:01 cron.notice crond[116]: USER manager pid 482 cmd nbqueue --
wipe
2006 Nov 10 13:55:01 cron.notice crond[116]: USER manager pid 483 cmd nbqueue --
wipe
.
.
.
```

Related Commands **show log config**
show log permanent

show log config

This command displays information about the logging system. This includes the configuration of the various log destinations, buffered, permanent, syslog servers (hosts) and email addresses. This also displays the latest status information for each of these destinations.

Syntax `show log config`

Mode User Exec, Privileged Exec and Global Configuration

Example To display the logging configuration use the command:

```
awplus# show log config
```

Output **Figure 12-5: Example output from the show log config command**

```
Buffered log:
Status ......... enabled
  Maximum size ... 100kb
  Filters:
 *1 Level ........ notices
    Program ...... any
    Facility ..... any
    Message text . any
  2 Level ........ informational
    Program ...... mstp
    Facility ..... daemon
    Message text . any
  Statistics ..... 1327 messages received, 821 accepted by filter (2006 Dec 11
10:36:16)
Permanent log:
  Status ......... enabled
  Maximum size ... 60kb
  Filters:
  1 Level ........ error
    Program ...... any
    Facility ..... any
    Message text . any
 *2 Level ........ warnings
    Program ...... dhcp
    Facility ..... any
    Message text . "pool exhausted"
  Statistics ..... 1327 messages received, 12 accepted by filter (2006 Dec 11
10:36:16)
Host 10.32.16.21:
  Time offset .... +2:00
  Offset type .... UTC
  Filters:
  1 Level ........ critical
    Program ...... any
    Facility ..... any
    Message text . any
  Statistics ..... 1327 messages received, 1 accepted by filter (2006 Dec 11
10:36:16)
Email admin@alliedtelesis.com:
  Time offset .... +0:00
  Offset type .... Local
  Filters:
  1 Level ........ emergencies
    Program ...... any
    Facility ..... any
    Message text . any
  Statistics ..... 1327 messages received, 0 accepted by filter (2006 Dec 11
10:36:16)
Monitor log:
  Filters:
 *1   Level ...... debugging
      Program .... any
      Facility ... any
      Msg text ... any
  Statistics ..... Not available
Console log:
  Status ......... enabled
  List of consoles:
  1 .............. ttyS0
  Filters:
 *1   Level ...... critical
      Program .... any
      Facility ... any
      Msg text ... any
  Statistics ..... 1327 messages received, 1 accepted by filter (2006 Dec 11
10:36:16)
```

In the above example the '*' next to filter 1 in the buffered log configuration indicates that this is the default filter. The permanent log has had its default filter removed, so none of the filters are marked with "*'.

Note Terminal log and console log cannot be set at the same time. If console logging is enabled then the terminal logging is turned off.

Related Commands

show counter log
show log
show log permanent

show log permanent

This command displays the contents of the permanent log.

When used within a stacked environment, this command will display the contents of the permanent log for all the stack members, unless you specify a particular stack member.

Syntax `show log permanent[<stack-ID>][tail [<10-250>]]`

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.
`tail`	Display only the latest log entries.
<10-250>	Specify the number of log entries to display.

Default If the optional `tail` parameter is specified only the latest 10 messages in the permanent log are displayed. A numerical value can be specified after the `tail` parameter to select how many of the latest messages should be displayed.

Mode User Exec, Privileged Exec and Global Configuration

Example To display the permanent log of stack member 2, use the command:

```
awplus# show log permanent 2
```

Output **Figure 12-6: Example output from the show log permanent command**

```
awplus#show log permanent 2

Stack member 2:

<date> <time> <facility>.<severity> <program[<pid>]>: <message>
-------------------------------------------------------------------------

2014 Feb 25 09:10:48 daemon.crit awplus-2 HPI: HOTSWAP Pluggable 2.0.51 hotswapped
in: AT-StackXS/1.0
2014 Feb 25 09:10:48 daemon.crit awplus-2 HPI: HOTSWAP Pluggable 2.0.52 hotswapped
in: 2127931-2
2014 Feb 25 09:10:50 user.crit awplus-2 VCS[922]: Member 1 (eccd.6d7d.a50e) has
joined the stack
2014 Feb 25 09:10:52 user.crit awplus-2 VCS[922]: Member 1 (eccd.6d7d.a50e) has
become the Active Master
2014 Feb 25 09:10:52 local6.alert awplus-2 VCS[922]: stack member has booted from
non-default location, SW version auto synchronization cannot be supported.
2014 Feb 25 09:10:52 user.crit awplus-2 VCS[922]: Stack Virtual MAC is
0000.cd37.0002
2014 Feb 25 09:11:46 user.crit awplus-2 ATMF[862]: awplus-x510 has joined. 1
member in total.
```

Related Commands show log

show running-config log

This command displays the current running configuration of the Log utility.

Syntax `show running-config log`

Mode Privileged Exec and Global Configuration

Example To display the current configuration of the log utility, use the command:

```
awplus# show running-config log
```

Related Commands **show log**
show log config

Chapter 13: Scripting Commands

Command List 13.2
activate 13.2
echo 13.3
wait 13.4

Command List

This chapter provides commands used for command scripts.

activate

This command activates a script file.

Syntax `activate [background] <script>`

Parameter	Description
`background`	Activate a script to run in the background. A process that is running in the background will operate as a separate task, and will not interrupt foreground processing. Generally, we recommend running short, interactive scripts in the foreground and longer scripts in the background. The default is to run the script in the foreground.
<script>	The file name of the script to activate. The script is a command script consisting of commands documented in this software reference. Note that you must use either a **.scp** or a **.sh** filename extension for a valid script text file, as described below in the usage section for this command.

Mode Privileged Exec

Usage When a script is activated, the privilege level is set to 1 enabling User Exec commands to run in the script. If you need to run Privileged Exec commands in your script you need to add an **enable (Privileged Exec mode)** command to the start of your script. If you need to run Global Configuration commands in your script you need to add a **configure terminal** command after the **enable** command at the start of your script.

The **activate** command executes the script in a new shell. A **terminal length** shell command, such as **terminal length 0** may also be required to disable a delay that would pause the display.

A script must be a text file with a filename extension of either **.sh** or **.scp** only for the AlliedWare Plus™ CLI to activate the script file. The **.sh** filename extension indicates the file is an ASH script, and the **.scp** filename extension indicates the file is an AlliedWare Plus™ script.

Examples To activate a command script to run as a background process, use the command:

```
awplus# activate background test.scp
```

Related Commands
configure terminal
echo
enable (Privileged Exec mode)
wait

echo

This command echoes a string to the terminal, followed by a blank line.

Syntax `echo <line>`

Parameter	Description
<line>	The string to echo

Mode User Exec and Privileged Exec

Usage This command may be useful in CLI scripts, to make the script print user-visible comments.

Example To echo the string `Hello World` to the console, use the command:

```
awplus# echo Hello World
```

Output

```
Hello World
```

Related Commands **activate**
wait

wait

This command pauses execution of the active script for the specified period of time.

Syntax `wait <delay>`

Parameter	Description
<delay>	<1-65335> Specify the time delay in seconds

Default No wait delay is specified by default to pause script execution.

Mode Privileged Exec (when executed from a script not directly from the command line)

Usage Use this command to pause script execution in an **.scp** (AlliedWare Plus™ script) or an **.sh** (ASH script) file executed by the **activate** command. The script must contain an **enable (Privileged Exec mode)** command since the **wait** command is only executed in the Privileged Exec mode.When a script is activated, the privilege level is set to 1 enabling User Exec commands to run in the script. If you need to run Privileged Exec commands in your script you need to add an **enable (Privileged Exec mode)** command to the start of your script.

Example See an example **.scp** script file extract below that will show port counters for interface `port1.0.1` over a 10 second interval:

```
enable
show interface port1.0.1
wait 10
show interface port1.0.1
```

Related Commands
activate
echo
enable (Privileged Exec mode)

Chapter 14: Interface Commands

Command List 14.2
description (interface) 14.2
interface (to configure) 14.3
mru 14.5
mtu 14.7
show interface 14.9
show interface brief 14.12
show interface status 14.13
shutdown 14.15

Command List

This chapter provides an alphabetical reference of commands used to configure and display interfaces.

description (interface)

Use this command to add a description to a specific port or interface.

Syntax `description <description>`

Parameter	Description
<description>	Text describing the specific interface.

Mode Interface Configuration

Example The following example uses this command to describe the device that a switch port is connected to.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# description Boardroom PC
```

interface (to configure)

Use this command to select one or more interfaces to configure.

Syntax

```
interface <interface-list>
interface lo
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface such as a VLAN (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.6`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`lo`	The local loopback interface.

Usage

A local loopback interface is one that is always available for higher layer protocols to use and advertise to the network. Although a local loopback interface is assigned an IP address, it does not have the usual requirement of connecting to a lower layer physical entity. This lack of physical attachment creates the perception of a local loopback interface always being accessible via the network.

Local loopback interfaces can be utilized by a number of protocols for various purposes. They can be used to improve access to the switch and also increase its reliability, security, scalability and protection. In addition, local loopback interfaces can add flexibility and simplify management, information gathering and filtering.

One example of this increased reliability is for OSPF to advertise a local loopback interface as an interface-route into the network irrespective of the physical links that may be “up” or “down” at the time. This provides a higher probability that the routing traffic will be received and subsequently forwarded.

Mode

Global Configuration

Example

The following example shows how to enter Interface mode to configure `vlan1`. Note how the prompt changes.

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)#
```

The following example shows how to enter Interface mode to configure the local loopback interface.

```
awplus# configure terminal
awplus(config)# interface lo
awplus(config-if)#
```

Related Commands

ip address
show interface
show interface brief

mru

Use this command to set the Maximum Receive Unit (MRU) size for switch ports, where MRU is the maximum frame size (excluding headers) that switch ports can receive. For more information, see **"Support for Jumbo Frames" on page 16.16**.

Use the **no** variant of this command to remove a previously specified Maximum Receive Unit (MRU) size for switch ports, and restore the default MRU size (1500 bytes) for switch ports.

Note

The figure of 1500 bytes specifies the payload only. For an IEEE 802.1q frame, provision is made (internally) for the following additional components:

- Source and Destination addresses
- EtherType field
- Priority and VLAN tag fields
- FCS

These additional components increase the frame size internally to 1522 bytes.

Syntax

```
mru <mru-size>
no mru
```

Parameter	Description
<mru-size>	<68-16357> Specifies the Maximum Receive Unit (MRU) size in bytes, where: ■ 1500 bytes is the default Ethernet MRU size for an interface.

Default The default MRU size is 1500 bytes for switch ports.

Mode Interface Configuration for switch ports.

Usage Note that **show interface** output will only show MRU size for switch ports.

Examples To configure an MRU of 16357 bytes on `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# mru 16357
```

To configure an MRU of 1500 bytes on `port1.0.2` to `port1.0.4` use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.4
awplus(config-if)# mru 1500
```

To restore the MRU size of 1500 bytes on `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no mru
```

Related Commands **show interface**

mtu

Use this command to set the Maximum Transmission Unit (MTU) size for VLANs, where MTU is the maximum packet size that VLANs can transmit. The MTU size setting is applied to both IPv4 and IPv6 packet transmission.

Use the **no** variant of this command to remove a previously specified Maximum Transmission Unit (MTU) size for VLANs, and restore the default MTU size (1500 bytes) for VLANs.

Syntax

```
mtu <mtu-size>
no mtu
```

Parameter	Description
<mtu-size>	<68-1500> Specifies the Maximum Transmission Unit (MTU) size in bytes, where 1500 bytes is the default Ethernet MTU size for an interface.

Default The default MTU size is 1500 bytes for VLAN interfaces.

Mode Interface Configuration for VLAN interfaces.

Usage If a switch receives an IPv4 packet for Layer 3 switching to another VLAN with an MTU size smaller than the packet size, and if the packet has the '**don't fragment**' bit set, then the switch will send an ICMP '**destination unreachable**' (3) packet type and a '**fragmentation needed and DF set**' (4) code back to the source. For IPv6 packets bigger than the MTU size of the transmitting VLAN interface, an ICMP '**packet too big**' (ICMP type 2 code 0) message is sent to the source.

Note that **show interface** output will only show MTU size for VLAN interfaces.

Examples To configure an MTU size of 1500 bytes on interface `vlan2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# mtu 1500
```

To configure an MTU size of 1500 bytes on interfaces `vlan2` to `vlan4`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# mtu 1500
```

To restore the MTU size to the default MTU size of 1500 bytes on `vlan2`, use the commands

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no mtu
```

To restore the MTU size to the default MTU size of 1500 bytes on `vlan2` and `vlan4`, use the commands

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# no mtu
```

Related Commands **show interface**

show interface

Use this command to display interface configuration and status.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface such as a VLAN (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.6`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`lo`	The local loopback interface.

Mode User Exec and Privileged Exec

Usage Note that the output displayed with this command will show MTU (Maximum Transmission Unit) size for VLAN interfaces, and MRU (Maximum Received Unit) size for switch ports.

Example To display configuration and status information for all interfaces, use the command:

```
awplus# show interface
```

Figure 14-1: Example output from the show interface command

```
awplus#show interface
Interface port1.0.1
  Scope: both
  Link is UP, administrative state is UP
  Thrash-limiting
    Status Not Detected, Action link-down, Timeout 60(s)
  Hardware is Ethernet, address is 0000.cd24.daeb
  index 5001 metric 1 mru 1500
  <UP,BROADCAST,RUNNING,MULTICAST>
  current duplex half, current speed 100
  configured duplex auto, configured speed auto, configured polarity auto
  current ecofriendly lpi
  configured ecofriendly lpi
  SNMP link-status traps: Sending (Suppressed after 20 traps in 60 sec.)
    input packets 2396, bytes 324820, dropped 0, multicast packets 2370
    output packets 73235, bytes 406566, multicast packets 7321 broadcast packets 7
  Time since last state change: 0 days 16:35:52

Interface port2.0.2
  Scope: both
  Link is DOWN, administrative state is UP
  Thrash-limiting
    Status Unknown, Action learn-disable, Timeout 1(s)
  Hardware is Provisioned, address is 0000.0000.0000
  index 8001 metric 1 mru 1500
  <BROADCAST,MULTICAST>
  current duplex half, current speed 100
  configured duplex auto, configured speed auto, configured polarity auto
  current ecofriendly lpi
  configured ecofriendly lpi
  SNMP link-status traps: Disabled
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 16:35:52


Interface lo
  Scope: both
  Link is UP, administrative state is UP
  Hardware is Loopback
  index 1 metric 1
  <UP,LOOPBACK,RUNNING>
  SNMP link-status traps: Disabled
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 16:35:52

Interface vlan1
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0000.cd24.daa8
  index 201 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  VRF Binding: Not bound
  SNMP link-status traps: Disabled
  Bandwidth 1g
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 29, bytes 1334, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 05:36:40
```

To display configuration and status information for interface `lo`, use the command:

```
awplus# show interface lo
```

Figure 14-2: Example output from the show interface lo command

```
awplus#show interface lo
Interface lo
  Scope: both
  Link is UP, administrative state is UP
  Hardware is Loopback
  index 1 metric 1
  <UP,LOOPBACK,RUNNING>
  SNMP link-status traps: Disabled
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
  Time since last state change: 69 days 01:28:47
```

To display configuration and status information for interfaces `vlan1` and `vlan2`, use the command:

```
awplus# show interface vlan1,vlan2
```

Figure 14-3: Example output from the show interface vlan1,vlan2 command

```
awplus#show interface vlan1,vlan2
Interface vlan1
  Scope: both
  Link is UP, administrative state is UP
  Hardware is VLAN, address is 0015.77e9.5c50
  IPv4 address 192.168.1.1/24 broadcast 192.168.1.255
  index 201 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,RUNNING,MULTICAST>
  SNMP link-status traps: Disabled
  Bandwidth 1g
    input packets 295606, bytes 56993106, dropped 5, multicast packets 156
    output packets 299172, bytes 67379392, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 14:22:39

 Interface vlan2
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0015.77e9.5c50
  IPv4 address 192.168.2.1/24 broadcast 192.168.2.255
  Description: ip_phone_vlan
  index 202 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  SNMP link-status traps: Disabled
  Bandwidth 1g
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 90, bytes 4244, multicast packets 0 broadcast packets 0
  Time since last state change: 0 days 14:22:39
```

Related Commands

- **ecofriendly lpi**
- **mru**
- **mtu**
- **show interface brief**

show interface brief

Use this command to display brief interface, configuration, and status information, including provisioning information.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show interface brief`

Mode User Exec and Privileged Exec

Output **Figure 14-4: Example output from the show interface brief command**

```
awplus#show int brief
Interface              Status          Protocol
port1.0.1              admin up        down
port1.0.2              admin up        down
port1.0.3              admin up        down
port1.0.4              admin up        down
port1.0.5              admin up        down
port1.0.6              admin up        running
lo                     admin up        running
vlan1                  admin up        down
vlan2                  admin up        down
```

Table 14-1: Parameters in the output of the show interface brief command

Parameter	Description
`Interface`	The name or type of interface.
`Status`	The administrative state. This can be either **admin up** or **admin down**.
`Protocol`	The link state. This can be either **down**, **running**, or **provisioned**.

Related Commands **show interface**
show interface memory

show interface status

Use this command to display the status of the specified interface or interfaces. Note that when no interface or interfaces are specified then the status of all interfaces on the switch are shown.

Syntax `show interface [<port-list>] status`

Parameter	Description
`<port-list>`	The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.6`) a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.6`, or `sa1-2`, or `po1-2` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix switch ports, static channel groups, and dynamic (LACP) channel groups in the same list

Examples To display the status of ports 1.0.1 to 1.0.5, use the commands:

```
awplus# show interface port1.0.1-1.0.4 status
```

Figure 14-5: Example output from the show interface <port-list> status command

```
awplus#show interface port1.0.1 -1.0.5 status
Port       Name               Status            Vlan Duplex   Speed Type
port1.0.1                     notconnect           1 auto      auto 1000BASE-T
port1.0.2                     notconnect           1 auto      auto 1000BASE-T
port1.0.3                     notconnect           1 auto      auto 1000BASE-T
port1.0.4                     notconnect           1 auto      auto 1000BASE-T
```

To display the status of all ports, use the commands:

```
awplus# show interface status
```

Figure 14-6: Example output from the show interface status command

```
awplus#sho int status
Port      Name               Status            Vlan Duplex   Speed Type
port1.0.1 Trunk_Net          connected        trunk a-full  a-1000 1000BaseTX
port1.0.2 Access_Net1        connected            5 full      100 1000BaseTX
port1.0.3 Access_Net1        disabled             5 auto      auto 1000BaseTX
port1.0.4 Access_Net2        connected            6 a-half   a-100 1000BaseTX
port1.0.5 Private_Prom       connected           10 a-full   a-100 1000BaseTX
port1.0.6 Private_Net1       connected        10,11 a-full   a-100 1000BaseTX
port1.0.7 Private_Net2       connected        10,12 a-full   a-100 1000BaseTX
port1.0.8                    notconnect           1 auto      auto 1000BaseTX
.
.
port1.0.23                   disabled             1 auto      auto not present
port1.0.24                   notconnect           1 auto      auto unknown
sa1                          notconnect       trunk auto      auto
```

Table 14-2: Parameters in the output from the show interface status command

Parameter	Description
`Port`	Name/Type of the interface.
`Name`	Description of the interface.
`Status`	The administrative and operational status of the interface; one of: ■ disabled: the interface is administratively down. ■ connect: the interface is operationally up. ■ notconnect: the interface is operationally down.
`Vlan`	VLAN type or VLAN IDs associated with the port: ■ When the VLAN mode is trunk, it displays **trunk** (it does not display the VLAN IDs). ■ When the VLAN mode is access, it displays the VLAN ID. ■ When the VLAN mode is private promiscuous, it displays the primary VLAN ID if it has one, and **promiscuous** if it does not have a VLAN ID. ■ When the VLAN mode is private host, it displays the primary and secondary VLAN IDs. ■ When the port is an Eth port, it displays **none**: there is no VLAN associated with it. ■ When the VLAN is dynamically assigned, it displays the current dynamically assigned VLAN ID (not the access VLAN ID), or **dynamic** if it has multiple VLANs dynamically assigned.
`Duplex`	The actual duplex mode of the interface, preceded by **a-** if it has autonegotiated this duplex mode. If the port is disabled or not connected, it displays the configured duplex setting.
`Speed`	The actual link speed of the interface, preceded by **a-** if it has autonegotiated this speed. If the port is disabled or not connected, it displays the configured speed setting.
`Type`	The type of interface, e.g. 1000BaseTX. For SFP bays, it displays **Unknown** if it does not recognize the type of SFP installed, or **Not present** if an SFP is not installed or is faulty.

Related Commands

show interface
show interface memory

shutdown

This command shuts down the selected interface. This administratively disables the link and takes the link down at the physical (electrical) layer.

Use the **no** variant of this command to disable this function and therefore to bring the link back up again.

Syntax

```
shutdown
no shutdown
```

Mode Interface Configuration

Example The following example shows the use of the `shutdown` command to shut down `port1.0.2`.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# shutdown
```

The following example shows the use of the `no shutdown` command to bring up `port1.0.2`.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no shutdown
```

The following example shows the use of the `shutdown` command to shut down `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# shutdown
```

The following example shows the use of the `no shutdown` command to bring up `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no shutdown
```

Chapter 15: Interface Testing Commands

Command List 15.2
clear test interface 15.2
service test 15.3
test interface 15.4

Command List

This chapter provides an alphabetical reference of commands used for testing interfaces.

clear test interface

This command clears test results and counters after issuing a test interface command. Test results and counters must be cleared to issue subsequent test interface commands later on.

Syntax `clear test interface {<port-list>|all}`

Parameter	Description
`<port-list>`	The ports to test. A port-list can be: ■ a switch port (e.g. `port1.0.6`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-port1.0.6` ■ a comma-separated list of the above, e.g. `port1.0.1,port1.0.5-1.0.6` The specified ports must exist.
`all`	All interfaces

Mode Privileged Exec

Examples To clear the counters for `port1.0.1` use the command:

```
awplus# clear test interface port1.0.1
```

To clear the counters for all interfaces use the command:

```
awplus# clear test interface all
```

Related Commands test interface

service test

This command puts the device into the interface testing state, ready to begin testing. After entering this command, enter Interface Configuration mode for the desired interfaces and enter the command **test interface**.

Do not test interfaces on a device that is part of a live network—disconnect the device first.

Use the **no** variant of this command to stop the test service.

Syntax

```
service test
no service test
```

Mode Global Configuration

Example To put the device into a test state, use the command:

```
awplus(config)# service test
```

Related Commands **test interface**

test interface

This command starts a test on a port or all ports or a selected range or list of ports.

Use the **no** variant of this command to disable this function. The test duration can be configured by specifying the time in minutes after specifying a port or ports to test.

For an example of all the commands required to test switch ports, see the Examples section in this command. To test the Eth port, set its speed to 100 by using the command **speed 100.**

Note Do not run test interface on live networks because this will degrade network performance.

Syntax

```
test interface {<port-list>|all} [time{<1-60>|cont}]
no test interface {<port-list>|all}
```

Parameter	Description
<port-list>	The ports to test. A port-list can be: ■ a switch port (e.g. `port1.0.6`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-port1.0.6` ■ a comma-separated list of the above, e.g. `port1.0.1,port1.0.5-1.0.6` The specified ports must exist.
`all`	All ports
`time`	Keyword entered prior to the value for the time duration of the interface test.
<1-60>	Specifies duration of time to test the interface or interfaces in minutes (from a minimum of 1 minute to a maximum of 60 minutes). The default is 4 minutes.
`cont`	Specifies continuous interface testing until canceled with command negation.

Mode Privileged Exec

Example To test the switch ports in VLAN 1, install loopbacks in the ports, and enter the following commands:

```
awplus(config)# service test
awplus(config)# no spanning-tree rstp enable bridge-forward
awplus(config)# interface vlan1
awplus(config-if)# shutdown
awplus(config-if)# end
awplus# test interface all
```

To see the output, use the commands:

```
awplus# show test
```

```
awplus# show test count
```

To start the test on all interfaces for 1 minute use the command:

```
awplus# test interface all time 1
```

Related Commands **clear test interface**

Part 2: Layer Two Switching

- Chapter 16 Switching Introduction
- Chapter 17 Switching Commands
- Chapter 18 VLAN Introduction
- Chapter 19 VLAN Commands
- Chapter 20 Spanning Tree Introduction: STP, RSTP, and MSTP
- Chapter 21 Spanning Tree Commands
- Chapter 22 Link Aggregation Introduction and Configuration
- Chapter 23 Link Aggregation Commands
- Chapter 24 Power over Ethernet Introduction
- Chapter 25 Power over Ethernet Commands
- Chapter 26 GVRP Introduction and Configuration
- Chapter 27 GVRP Commands

Chapter 16: Switching Introduction

Introduction 16.2
Physical Layer Information 16.3
Switch Ports 16.3
Activating and Deactivating Switch Ports 16.4
Autonegotiation 16.4
Duplex Mode 16.4
Speed Options 16.4
MDI/MDIX Connection Modes 16.5
The Layer 2 Switching Process 16.7
The Ingress Rules 16.7
The Learning Process 16.8
The Forwarding Process 16.9
The Egress Rules 16.9
Layer 2 Filtering 16.11
Ingress Filtering 16.11
Storm Control 16.12
Loop Protection 16.13
Loop Detection 16.13
Thrash Limiting 16.14
Support for Jumbo Frames 16.16
Port Mirroring 16.17
Port Security 16.18
MAC Address Learn Limits 16.18
IEEE 802.1X 16.18
Static MAC Address 16.18
Quality of Service 16.19
IGMP Snooping 16.20

Introduction

This chapter gives an overview of Layer 1 and 2 switching.

Layer 2 switches are used to connect multiple Local Area Network (LAN) segments together to form an extended LAN. Stations connected to different LANs can be configured to communicate with one another as if they were on the same LAN. They can also divide one physical LAN into multiple Virtual LANs (VLANs). Stations connected to each other on the same extended LAN can be grouped in separate VLANs, so that a station in one VLAN can communicate directly with other stations in the same VLAN, but must go through higher layer routing protocols to communicate with those stations in other VLANs.

Layer 2 switches appear transparent to higher layer protocols, transferring frames between the data link layers of the networks to which they are attached. A Layer 2 switch accesses each physical link according to the rules for that particular network. Access may not always be instant, so the switch must be capable of storing and forwarding frames.

Storing and forwarding enables the switch to examine both the VLAN tag fields and Ethernet MAC address fields in order to forward the frames to their appropriate destination. In this way, the switch can act as an intelligent filtering device, redirecting or blocking the movement of frames between networks.

Because switch ports can sometimes receive frames faster than it can forward them, the switch has Quality of Service (QoS) queues in which frames await transmission according to their priority. Such a situation could occur where data enters a number of input ports all destined for the same output port.

The switch can be used to:

- Increase both the physical extent and the maximum number of stations on a LAN. LANs are limited in their physical extent by the signal distortion and propagation delay characteristics of the media. The switch overcomes this limitation by receiving a frame on one LAN and then retransmitting it to another. The physical characteristics of the LAN media also place a practical limit on the number of stations that can be connected to a single LAN segment. The switch overcomes this limitation by joining LAN segments to form an extended LAN capable of supporting more stations than either of the individual LAN segments.
- Connect LANs that have a common data link layer protocol but different physical media, for example, Ethernet 10BASET, 100BASET, and 10BASEF.
- Increase the availability of LANs by allowing multiple redundant paths to be physically configured and selected dynamically, using the Spanning Tree algorithm.
- Reduce the load on a LAN or increase the effective bandwidth of a LAN, by filtering traffic.
- Prioritize the transmission of data with high Quality of Service requirements.

By using Virtual LANs (VLANs), a single physical LAN can be separated into multiple Virtual LANs. VLANs can be used to:

- Further improve LAN performance, as broadcast traffic is limited to LAN segments serving members of the VLAN to which the sender belongs.
- Provide security, as frames are forwarded to those stations belonging to the sender's VLAN, and not to stations in other VLANs on the same physical LAN.
- Reduce the cost of moving or adding stations to function or security based LANs, as this generally requires only a change in the VLAN configuration.

Physical Layer Information

Switch Ports

A unique port number identifies each switch port. The software supports a number of features at the physical level that allow it to be connected in a variety of physical networks. This physical layer (Layer 1) versatility includes:

- Enabling and disabling of ports
- Auto negotiation of port speed and duplex mode, where supported by the port type
- Manual setting of port speed and duplex mode, where supported by the port type
- Link up and link down triggers
- Packet storm protection
- Port mirroring
- Support for SNMP management

Port numbering

Ports are numbered using a 3 digit format `x.y.z` where `x` is the device number (within a stacked configuration), `y` is the module number within the device, and `z` is the port number within the module. Ports connected directly to the switch chassis or baseboard (rather than a pluggable module) are given the module number `0`. In an unstacked configuration all device numbers are `1`. For example, `port1.2.6` represents device `1`, module `2`, port `6`.

Adding a description

You can add a description to an interface to help identify its purpose or position. For example, to add the description "connected to Nerv" to `port1.0.3`, use the commands:

```
awplus(config)# interface port1.0.3
awplus(config-if)# description connected to Nerv
```

Port ranges

Continuous

To configure a continuous range of ports at the same time, enter the range in the format:

`portx.y.z-portx.y.z`

For example, to configure the same interface setting on `port1.0.1` to `port1.0.2`, enter the Global Configuration mode command:

```
awplus(config)# interface port1.0.1-port1.0.2
```

Non-continuous

To configure a non-continuous set of ports at the same time, enter a comma-separated list:

`portx.y.z,portx.y.z`

For example, to configure the same interface setting on `port1.0.1` and `port1.0.5`, enter the Global Configuration mode command:

```
awplus(config)# interface port1.0.1,port1.0.5
```

You can combine a hyphen-separated range and a comma-separated list. To configure the same setting on `port1.0.1` to `port1.0.3` and `port1.0.5`, enter the Global Configuration mode command:

```
awplus(config)# interface port1.0.1-port1.0.3,port1.0.5
```

Activating and Deactivating Switch Ports

An active switch port is one that is available for packet reception and transmission. Disabling a switch port does not affect the STP operation on the port. By default ports and VLANs are activated.

To shutdown a port or VLAN use the **shutdown** command on page 14.15. Use the **no** variant of this command to reactivate it.

Autonegotiation

Autonegotiation lets the port adjust its speed and duplex mode to accommodate the device connected to it. When the port connects to another autonegotiating device, they negotiate the highest possible speed and duplex mode for both of them.

By default, all ports autonegotiate. Setting the port to a fixed speed and duplex mode may be necessary when connecting to a device that cannot autonegotiate.

Duplex Mode

Ports can operate in full duplex or half duplex mode depending on the type of port it is. When in full duplex mode, a port transmits and receives data simultaneously. When in half duplex mode, the port transmits or receives but not both at the same time.

You can set a port to use either of these options, or allow it to autonegotiate the duplex mode with the device at the other end of the link. To configure the duplex mode, use these commands:

`awplus#` `configure terminal`	Enter Global Configuration mode
`awplus(config)#` `interface port1.0.1`	Enter Interface Configuration mode for port 1.0.1
`awplus(config-if)#` `duplex {auto|full|half}`	Enter the Duplex mode for port 1.0.1

Speed Options

Before configuring a port's speed, check the hardware limit for the particular port type. The following list can be used as a guide:

- non-SFP RJ-45 copper switch ports: 10, 100 or 1000 Mbps
- supported tri-speed copper SFPs: 10, 100 or 1000 Mbps
- fiber SFPs: 100 Mbps to 1000 Mbps, depending on the SFP type
- SFP+ / XFP modules: 10 Gbps

For the latest list of approved SFP transceivers either contact your authorized distributor or reseller, or visit **alliedtelesis.com**.

You can set a port to use one of these speed options, or allow it to autonegotiate the speed with the device at the other end of the link.

Most types of switch port can operate in either full duplex or half duplex mode. In full duplex mode a port can transmit and receive data simultaneously. In half duplex mode the port can either transmit or receive, but not at the same time.

Make sure that the configuration of the switch matches the configuration of the device at the far end of the link. In particular, avoid having one end autonegotiate duplex mode while the other end is fixed. For example, if you set one end of a link to autonegotiate and fix the other end at full duplex, the autonegotiating end cannot determine that the fixed end is full duplex capable. Therefore, the autonegotiating end selects half-duplex operation. This results in a duplex mismatch and packet loss. To avoid this, either fix the mode at both ends, or use autonegotiation at both ends.

Configuring the port speed

To set the port speed to 1000 Mbps on port 1.0.1, use the commands:

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`interface port1.0.1`	Enter Interface Configuration mode for port 1.0.1
`awplus(config-if)#`	
`speed 1000`	Set the port speed for port 1.0.1 to 1000 Mbps.

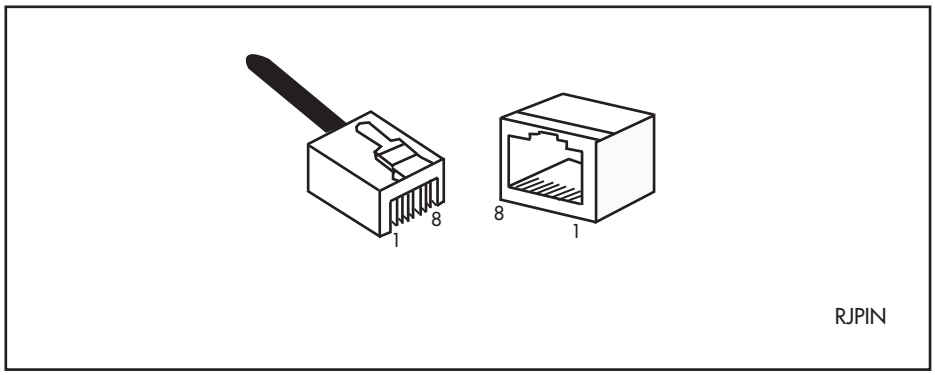
MDI/MDIX Connection Modes

By default, copper 10Base-T, 100Base-T, and 1000Base-T ports on the switch automatically set the Media Dependant Interface mode to MDI or MDIX for successful physical connections. We recommend using this default setting. However, you can configure them to have either fixed MDI mode or fixed MDIX mode by using the **polarity** command on page 17.29. MDI/MDIX mode polarity does not apply to fiber ports.

Connections to 10BASE-T, 100BASE-T, and 1000BASE-T networks may either be straight though (MDI) or crossover (MDIX). The crossover connection can be achieved by using either a crossover cable or by integrating the crossover function within the device. In the latter situation, the connector is referred to as an MDIX connection. Refer to your switch's Hardware Reference for more detailed information on physical connections cabling.

The IEEE 802.3 standard defines a series of Media Dependant Interface types and their physical connections. For twisted pair (10BASE-T) networking, the standard defines that connectors that conform to the IEC 60603-7 standard. The **Figure 16-1**f shows a connector of this type.

Figure 16-1: Connector used for 10BASE-T networks



The Layer 2 Switching Process

The Layer 2 switching process comprises these related but separate processes:

- **The Ingress Rules**
- **The Learning Process**
- **The Forwarding Process**
- **The Egress Rules**

Ingress rules admit or discard frames based on their VLAN tagging.

The Learning process learns the MAC addresses and VLAN membership of frames admitted on each port.

The Forwarding process determines which ports the frames are forwarded to, and the Quality of Service priority with which they are transmitted.

Finally, Egress rules determine for each frame whether VLAN tags are included in the Ethernet frames that are transmitted.

These processes assume that each station on the extended LAN has a unique data link layer address, and that all data link layer frames have a header which includes the source (sender's) MAC address and destination (recipient's) MAC address.

The Ingress Rules

All frames, tagged and untagged, that a VLAN-aware switch receives must be classified into a VLAN. Each received frame is mapped to exactly one VLAN. If an incoming frame is tagged with a valid VLAN identifier (VID) then that VID is used. If an incoming frame is untagged or is priority tagged (a tagged frame with a VID of all zeros), then the switch uses internal VLAN association rules to determine the VLAN it belongs to. The default settings for the ingress rules are to Admit All Frames, and for Ingress Filtering to be on.

Every port belongs to one or more VLANs so every incoming frame has a VID to show which VLAN it belongs. The final part of the Ingress Rules depends on whether Ingress Filtering is enabled for the port. If Ingress Filtering is disabled, all frames are passed on to the Learning process, regardless of which VLAN they belong to. If Ingress Filtering is enabled (by default), frames are admitted only when they have the VID of a VLAN to which the port belongs. Frames are discarded when they do not have an associated VID matching the VLAN assigned to a port.

The possible association rules, in order of precedence, are:

- IP subnet/IPX network classification
- protocol classification
- port classification

The default VLAN classification is based upon the port on which the incoming frame (untagged, or priority tagged) was received. It is possible for an incoming untagged, or priority tagged, frame to match more than one of the association rules.

Each port on the switch can be configured to be one of two modes:

- only untagged frames - access mode
- VLAN-tagged frames - trunk mode

Access mode

This mode can be used to connect to VLAN unaware devices. Frames to and from access mode ports carry no VLAN tagging information.

Trunk mode

This mode is used to connect VLAN capable devices. All devices that connect using trunk mode ports must be VLAN aware.

You can specify a "native" VLAN on a trunk port. When the port receives untagged packets, it will tag the packets with the VID of the native VLAN. Note that packets from the native VLAN egress the port untagged.

To specify the native VLAN, use the **switchport trunk native vlan** command on page 19.30.

The Learning Process

The learning process uses an adaptive learning algorithm, sometimes called ***backward learning***, to discover the location of each station on the extended LAN.

All frames admitted by the ingress rules on any port are passed on to the forwarding process when they are for destinations in the same VLAN. Frames destined for other VLANs are passed to a Layer 3 protocol, such as IP. For every frame admitted, the frame's source MAC address and VID are compared with entries in the forwarding database for the VLAN (also known as a ***MAC Address table***) maintained by the switch. When the frame's source address is not in the forwarding database for the VLAN, the address is added and an ageing timer for that entry is started. When the frame's source address is already in the forwarding database, the ageing timer for that entry is restarted.

By default, switch learning is enabled. It can be disabled with the **no mac address-table acquire** command, and re-enabled using the **mac address-table acquire** command on page 17.18.

If the ageing timer for an entry in the forwarding database expires before another frame with the same source address is received, the entry is removed from the forwarding database. This prevents the forwarding database from being filled with information about stations that are inactive or have been disconnected from the network. It also ensures that entries for active stations are kept alive in the forwarding database.

By default, the ageing timer is enabled with a default ageing-time. The ageing timer can be reset to the default with the **no mac address-table ageing-time** command. The ageing timer can be increased or decreased using the **mac address-table ageing-time** command.

If switch learning is disabled and the ageing timer has aged out all dynamically learned filter entries, only statically entered MAC source addresses decide the packets to forward or discard. When the switch finds no matching entries in the forwarding database during the forwarding process, all switch ports in the VLAN are flooded with the packet, except the port that received it.

The default for the mac address-table ageing-time is 300 seconds (5 minutes) and can be modified by using the command **mac address-table ageing-time**. The **no mac address-table ageing-time** command will reset the ageing-time back to the default (5 minutes).

To set the mac address-table ageing-time to 1000 seconds:

`awplus#`	
`configure terminal`	Enter the config terminal mode
`awplus(config)#`	
`mac address-table ageing-time 1000`	Set the ageing time to 1000 seconds

To display general switch settings, including settings for switch learning and the switch ageing timer, use the **show system** command on page 10.47.

The Forwarding Process

After a VID is assigned to a frame using the ingress rules, the switch forwards it to the destination MAC address specified in the frame. To do this the switch must learn which MAC addresses are available on each port for each VLAN. When the destination MAC address is not found, the switch floods the frame on all ports that are members of the VLAN except the port on which the frame was received.

The forwarding database (also known as the ***MAC Address table***) determines the egress port on which the destination MAC address has been learned. MAC addresses are learned dynamically as part of the Layer 2 switching process.

The forwarding database is ordered according to MAC address and VLAN identifier. This means a MAC address can appear more than once in the forwarding database having been learned on the same port but for different VLANs. This could occur if the IP address of an end station is changed thereby moving the end station to a different IP subnet-based VLAN while still connected to the same switch port. When the forwarding database ageing process is enabled, old entries in the forwarding database are deleted after a user-configurable period.

If the destination address is found, the switch discards the frame when the port is not in the STP forwarding or disabled state if the destination address is on the same port as the source address, or if there is a static filter entry for the destination address set to **discard** (see **"Layer 2 Filtering" on page 16.11**). Otherwise, the frame is forwarded on the indicated port.

Forwarding occurs only when the port on which the frame was received is in the Spanning Tree forwarding or disabled state. The destination address is then looked up in the forwarding database for the VLAN.

The Egress Rules

After the forwarding process has determined from which ports and transmission queues to forward a frame, the egress rules for each port determine whether the outgoing frame is VLAN-tagged with its numerical VLAN identifier (VID).

A port must belong to a VLAN at all times unless the port has been set as the mirror port for the switch.

A port can transmit VLAN-tagged frames for any VLAN to which the port belongs. A port can transmit untagged frames for any VLAN for which the port is configured, e.g. IP subnet-based or protocol-based, unless prevented by the port-based VLAN egress rules. A port that belongs to a port-based VLAN can transmit untagged packets for only one VLAN. For more information about VLANs and VLAN tagging, see **Chapter 18, VLAN Introduction**.

For more information on port tagging see the following commands:

switchport mode access command on page 19.18

switchport mode trunk command on page 19.24

Layer 2 Filtering

The switch has a forwarding database (also known as the ***MAC address table***) whose entries determine whether frames are forwarded or discarded over each port. Entries in the forwarding database are created dynamically by the learning process. A dynamic entry is automatically deleted from the forwarding database when its ageing timer expires.

The forwarding database supports queries by the forwarding process as to whether frames with given values of the destination MAC address field should be forwarded to a given port.

For each VLAN, the destination MAC address of a frame to be forwarded is checked against the forwarding database. If there is no entry for the destination address and VLAN, the frame is transmitted on all ports in the VLAN that are in the forwarding state, except the port on which the frame was received. This process is referred to as ***flooding***. If an entry is found in the forwarding database but the entry is not marked ***forwarding*** or the entry points to the same port the frame was received on, the frame is discarded. Otherwise, the frame is transmitted on the port specified by the forwarding database.

Ingress Filtering

The **ingress-filter** parameter of the **switchport mode trunk** command on page 19.24 and the **switchport mode access** command on page 19.18, enables or disables ingress filtering of frames entering the specified port (or port range).

Each port on the switch belongs to one or more VLANs. If ingress filtering is enabled, any frame received on the specified port is only admitted if its VID matches one for which the port is tagged. Any frame received on the port is discarded if its VID does not match one for which the port is tagged.

Untagged frames are admitted and are assigned the VLAN Identifier (VID) of the port's native VLAN. Ingress filtering can be turned off by setting the **disable** parameter of the above two commands. The default setting of the **enable** / **disable** parameter option is **enable**.

Note

Enabling the **vlan-disable** parameter of the **thrash-limiting** command on page 17.57 will also enable ingress filtering, and will override the setting of the switchport mode access, and trunk commands

Storm Control

The packet storm-control feature enables you to set limits on the reception rate of broadcast, multicast frames and destination lookup failures. You can set separate limits beyond which each of the different packet types are discarded.

Note A destination lookup failure (DLF) is the event of receiving a unicast Ethernet frame with an unknown destination address.

For more information on applying storm-control, see the **storm-control level** command on page 17.52.

To apply storm-control by limiting broadcasts to 30% on `port1.0.4`:

`awplus(config-if)#` `configure terminal`	Enter Global Configuration mode.
`awplus(config-if)#` `interface port1.0.4`	Enter the Interface Configuration mode for the selected port.
`awplus(config-if)#` `storm-control broadcast level 30`	Configure the interface.

To turn off storm protection on `port1.0.4`:

`awplus(config-if)#` `configure terminal`	Enter Global Configuration mode.
`awplus(config-if)#` `interface port1.0.4`	Enter the Interface Configuration mode for the selected port.
`awplus(config-if)#` `no storm-control broadcast level`	Configure the interface.

Loop Protection

Loop protection is a general term that embraces several different methods you can apply to protect your network from effects such as broadcast storms that can result from data loops or equipment malfunction.

Two methods of loop protection are available:

- **Loop Detection**
- **Thrash Limiting**

Loop Detection

Introduction

This feature is used to detect loops with a network segment. If a loop is detected then a selected protection mechanism is applied to limit the effect of the loop. The loop protection actions can be applied either to the port at which the loop is detected or to the VLAN within which the loop was detected.

Limiting Actions

You can configure loop detection to apply one of the following mechanisms when a loop condition is detected:

- Block all traffic on the port (or aggregated link) that detected the loop, and take **down** the link.
- Block all traffic on the port (or aggregated link) that detected the loop, but keep the link in the **up** state.
- Block all traffic on a vlan. Note that setting this parameter will also enable ingress filtering. This is the default action.
- Take no action, but log the details.
- Take no action.

Operation

To detect loops this feature operates by transmitting a series of Loop Detection Frames (LDFs) from each switch port out into the network. If no loops exist, then none of these frame should ever return. If a frame returns to its original port, the detection mechanism assumes that there is a loop somewhere in the network and offers a number of protective options.

Each LDF is a Layer 2 LLC frame that contains the following components:

- the source MAC address of the originating switch
- the destination MAC address of the non-existent end station 00-00-F4-27-71-01
- VLAN ID (where the port is a tagged member of a VLAN).
- a randomly generated LDF ID number.

You can set the detection mechanism to remember the LDF ID of up to 5 of the most recently transmitted LDF frames. Each of the 5 most recently transmitted frames is compared with every frame that arrives at that same port.

Configuration

To enable loop protection and configure its basic parameters, you use the **loop-protection** command on page 17.15.

Example To enable the loop-detect mechanism, and generate loop-detect frames once every 5 seconds, use the command:

```
awplus(config)# loop-protection loop-detect ldf-interval 5
```

Note LDFs are sent sequentially for each VLAN defined to a particular port. For example, if a particular port in this example is a member of 4 VLANs, then the LDFs will be sent from this port at the rate of 4 frames every 5 seconds.

You can now use the **loop-protection action** command on page 17.16 configure the action that the switch will take if a loop is detected.

Example To disable an interface, and bring the link down, when a network loop is detected, use the command:

```
awplus(config-if)# loop-protection action link-down
```

Now decide how long you want the protective action to apply for. You configure this function by using the **loop-protection timeout** command on page 17.17.

Example To configure a loop protection action timeout of 10 seconds, use the command:

```
awplus(config-if)# loop-protection timeout 10
```

Thrash Limiting

MAC address thrashing occurs when MAC addresses move rapidly between one or more ports or trunks, for example, due to a network loop.

Thrash limiting enables you to apply actions to a port when thrashing is detected. It is supported on all port types and also on aggregated ports.

Limiting Actions There are several different thrash actions that you can apply to a port when thrashing is detected. These actions are:

- **learnDisable:** MAC address learning is temporarily disabled on the port.
- **portDisable:** The port is logically disabled. Traffic flow is prevented, but the link remains up. The device at the other end does not notice that the port has changed status, and the link LEDs at both ends stay on.
- **linkDown:** The port is physically disabled and the link is down. This is equivalent to entering the **shutdown** command on page 14.15.
- **vlanDisable:** The port is disabled only for the VLAN on which thrashing has occurred. It can still receive and transmit traffic for any other VLANs of which it is a member.

When a MAC address is thrashing between two ports, one of these ports (the first to cross its thrashing threshold) is disabled. All other ports on the device will then have their threshold counters reset.

To set a thrash action for a port, use the **thrash-limiting** command on page 17.57:

To view the thrash action that is set for a port, use the **show interface switchport** command on page 17.32:

Re-enabling a port

When a port is disabled, either completely or for a specific VLAN, it remains disabled until it is manually re-enabled in any of the following ways:

- by using SNMP
- by rebooting the switch or stack
- by specifying a thrash timeout value along with the thrash action
- via the CLI

Support for Jumbo Frames

Jumbo frames are frames with more than 1500 bytes of payload. You can enable jumbo frame support on the switch to improve throughput and network utilization. Jumbo frame support allows you to put more data in each packet that the switch has to process.

The maximum received packet size is 16357 bytes.

You can increase the Maximum Receive Unit (MRU) size for switch ports to receive jumbo frames with payload larger than 1500 bytes. To increase MRU size, use the **mru** command on page 14.5.

Note

Jumbo packet switching is supported for L2 and L3 traffic flows that have established traffic paths. Jumbo frames cannot be used for establishing these network paths. Please use frames less than 1518 bytes to establish the path, for example by pinging the destination.

Port Mirroring

Port mirroring enables traffic being received and transmitted on a switch port to be sent to another switch port, the mirror port, usually for the purposes of capturing the data with a protocol analyzer.

The mirror port is the only switch port that does not belong to a VLAN, and therefore does not participate in any other switching. Before the mirror port can be set, it must be removed from all trunk groups and all VLANs except the default VLAN.

The following example sets mirroring on ports 1.0.2 and 1.0.5 for both incoming and outgoing data.

Note

Due to the internal hardware properties of the switch, frames that are destined to leave the mirrored port untagged (i.e. will have their VLAN tag removed on egress) will be received by the mirror port with the tag retained. Consequently, if frames were being transmitted by the mirror port (into the network) at wire speed, then the mirror port might be unable to accept all the frames supplied to it.

To configure port 1.0.2 to mirror port 1.0.5:

`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `interface port1.0.2`	Enter the Interface Configuration mode for port1.0.2.
`awplus(config-if)#` `mirror interface port1.0.5 direction both`	Configure this port to mirror port 1.0.5.

Port Security

The port security features provide control over the stations connected to each switch port.

These comprise:

- MAC address learn limits
- IEEE 802.1X
- Static MAC address

MAC Address Learn Limits

MAC address limiting is applied using the **switchport port-security** command on page 17.53. If enabled on a port, the switch will learn MAC addresses up to a user-defined limit from 1 to 256, then lock out all other MAC addresses. One of the following options can be specified for the action taken when an unknown MAC address is detected on a locked port:

- Discard the packet and take no further action.
- Discard the packet and notify management with an SNMP trap.
- Discard the packet, notify management with an SNMP trap and disable the port.

IEEE 802.1X

IEEE 802.1X restricts unauthenticated devices from connecting to the switch. After authentication is successful, traffic is allowed through the switch. For more information see **Chapter 68, 802.1X Introduction and Configuration**.

Static MAC Address

If a new MAC address is seen on a port with port security enabled and the MAC address is statically configured for another port, a violation is triggered. The maximum learn limit will be ignored and the specified intrusion action for the port will be carried out.

Quality of Service

Quality of Service (QoS) enables you to both prioritize traffic and limit its available bandwidth. The concept of QoS is a departure from the original networking protocols, in which all traffic on the Internet or within a LAN had the same available bandwidth. Without QoS, all traffic types are equally likely to be dropped if a link becomes oversubscribed. This approach is now inadequate in many networks, because traffic levels have increased and networks often carry time-critical applications such as streams of real-time video data. QoS also enables service providers to easily supply different customers with different amounts of bandwidth.

Configuring Quality of Service involves two separate stages:

1. Classifying traffic into flows, according to a wide range of criteria. Classification is performed by the switch's class maps.

2. Acting on these traffic flows.

For more information on QoS see **Chapter 62, Quality of Service (QoS) Introduction** and **Chapter 63, QoS Commands**.

IGMP Snooping

IGMP (Internet Group Management Protocol) is used by IP hosts to report their multicast group memberships to routers and switches. IP hosts join a multicast group to receive broadcast messages directed to the multicast group address. IGMP is an IP-based protocol and uses IP addresses to identify both the multicast groups and the host members. For a VLAN-aware devices, this means multicast group membership is on a per-VLAN basis. If at least one port in the VLAN is a member of a multicast group, by default multicast packets will be flooded onto all ports in the VLAN.

IGMP snooping enables the switch to forward multicast traffic intelligently on the switch. The switch listens to IGMP membership reports, queries and leave messages to identify the switch ports that are members of multicast groups. Multicast traffic will only be forwarded to ports identified as members of the specific multicast group.

IGMP snooping is performed at Layer 2 on VLAN interfaces automatically. By default, the switch will forward traffic only from those ports with multicast listeners, therefore it will not act as a simple hub and flood all multicast traffic out all ports. IGMP snooping is independent of the IGMP and Layer 3 configuration, so an IP interface does not have to be attached to the VLAN, and IGMP does not have to be enabled or configured.

IGMP snooping is enabled by default.

For more information on IGMP see **Chapter 48, IGMP and IGMP Snooping Introduction** and **Chapter 49, IGMP and IGMP Snooping Commands**.

Chapter 17: Switching Commands

Command List 17.2
backpressure 17.2
clear loop-protection counters 17.3
clear mac address-table static 17.4
clear mac address-table dynamic 17.5
clear port counter 17.7
debug loopprot 17.8
debug platform packet 17.9
duplex 17.11
flowcontrol (switch port) 17.12
linkflap action 17.14
loop-protection 17.15
loop-protection action 17.16
loop-protection timeout 17.17
mac address-table acquire 17.18
mac address-table ageing-time 17.19
mac address-table static 17.20
mac address-table thrash-limit 17.21
mirror interface 17.22
platform hwfilter-size 17.24
platform load-balancing 17.25
platform stop-unreg-mc-flooding 17.26
platform vlan-stacking-tpid 17.28
polarity 17.29
show debugging loopprot 17.30
show debugging platform packet 17.30
show flowcontrol interface 17.31
show interface switchport 17.32
show loop-protection 17.33
show mac address-table 17.34
show mac address-table thrash-limit 17.36
show mirror 17.37
show mirror interface 17.38
show platform 17.39
show platform classifier statistics utilization brief 17.40
show platform port 17.41
show port-security interface 17.47
show port-security intrusion 17.48
show storm-control 17.49
speed 17.50
storm-control level 17.52
switchport port-security 17.53
switchport port-security aging 17.54
switchport port-security maximum 17.55
switchport port-security violation 17.56
thrash-limiting 17.57
undebug loopprot 17.58
undebug platform packet 17.58

Command List

This chapter provides an alphabetical reference of commands used to configure switching. For more information see **Chapter 16, Switching Introduction**.

backpressure

This command provides a method of applying flow control to ports running in half duplex mode. The setting will only apply when the link is in the half-duplex state.

You can disable backpressure on an interface using the **off** parameter or the **no** variant of this command.

Syntax

```
backpressure {on|off}
no backpressure
```

Parameters	Description
`on`	Enables half-duplex flow control.
`off`	Disables half-duplex flow control.

Default Backpressure is turned off by default. You can determine whether an interface has backpressure enabled by viewing the running-config output; **backpressure on** is shown for interfaces if this feature is enabled.

Mode Interface Configuration

Usage The backpressure feature enables half duplex Ethernet ports to control traffic flow during congestion by preventing further packets arriving. Back pressure utilizes a pre-802.3x mechanism in order to apply Ethernet flow control to switch ports that are configured in the half duplex mode.

The flow control applied by the **flowcontrol (switch port)** command on page 17.12 operates only on full-duplex links, whereas back pressure operates only on half-duplex links.

If a port has insufficient capacity to receive further frames, the switch will simulate a collision by transmitting a CSMACD jamming signal from this port until the buffer empties. The jamming signal causes the sending switch to stop transmitting and wait a random period of time, before retransmitting its data, thus providing time for the buffer to clear. Although this command is only valid for switch ports operating in half-duplex mode the remote switch (the one sending the data) can be operating in the full duplex mode.

To see the currently-negotiated duplex mode for ports whose links are up, use the command **show interface**. To see the configured duplex mode (when different from the default), use the command **show running-config**.

Examples To enable back pressure flow control on interfaces `port1.0.1-port1.0.2` enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.2
awplus(config-if)# backpressure on
```

To disable back pressure flow control on interface `port1.0.2` enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# backpressure off
```

Validation Commands **show running-config**
show interface

Related Commands **duplex**

clear loop-protection counters

Use this command to clear the counters for the Loop Protection counters.

Syntax

```
clear loop-protection [interface <port-list>] counters
```

Parameters	Description
`interface`	The interface whose counters are to be cleared.
`<port-list>`	A port, a port range, or an aggregated link.

Mode Privileged Exec

Examples To clear the counter information:

```
awplus# clear loop-protection counters
awplus# clear loop-protection interface port1.0.1 counters
```

clear mac address-table static

Use this command to clear the filtering database of all statically configured entries for a selected MAC address, interface, or VLAN.

Syntax

```
clear mac address-table static [address <mac-address>|
   interface <port>|vlan <vid>]
```

Parameter	Description
`address`	Specify a MAC (Media Access Control) address to be cleared from the filtering database.
<mac-address>	Enter a MAC address to be cleared from the database in the format HHHH.HHHH.HHHH.
`interface`	Specify a switch port to be cleared from the filtering database.
<port>	Specify the switch port from which address entries will be cleared. This can be a single switch port, (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).
`vlan`	Specify a VLAN to be cleared from the filtering database.
<vid>	Enter a VID (VLAN ID) in the range `<1-4094>` to be cleared from the filtering database.

Mode Privileged Exec

Usage Use this command with options to clear the filtering database of all entries made from the CLI for a given MAC address, interface or VLAN. Use this command without options to clear any entries made from the CLI.

Compare this usage with **clear mac address-table dynamic** command on page 17.5.

Examples This example shows how to clear all filtering database entries configured through the CLI.

```
awplus# clear mac address-table static
```

This example shows how to clear all filtering database entries for a given interface configured through the CLI.

```
awplus# clear mac address-table static interface port1.0.3
```

This example shows how to clear filtering database entries filtering database entries configured through the CLI for a given mac address.

```
awplus# clear mac address-table static address 0202.0202.0202
```

Related Commands
clear mac address-table dynamic
mac address-table static
show mac address-table

clear mac address-table dynamic

Use this command to clear the filtering database of all entries learned for a selected MAC address, an MSTP instance, a switch port interface or a VLAN interface.

Syntax

```
clear mac address-table dynamic [address <mac-address>|
   interface <port> [instance <inst>]|vlan <vid>]
```

Parameter	Description
`interface`	Specify a switch port to be cleared from the filtering database.
<port>	Specify the switch port from which address entries will be cleared. This can be a single switch port, (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).
`address`	Specify a MAC (Media Access Control) address to be cleared from the filtering database.
<mac-address>	Enter a MAC address to be cleared from the database in the format HHHH.HHHH.HHHH.
`instance`	Specify an MSTP (Multiple Spanning Tree) instance to be cleared from the filtering database.
<inst>	Enter an MSTP instance in the range `<1-63>` to be cleared from the filtering database.
`vlan`	Specify a VLAN to be cleared from the filtering database.
<vid>	Enter a VID (VLAN ID) in the range `<1-4094>` to be cleared from the filtering database.

Mode Privileged Exec

Usage Use this command with options to clear the filtering database of all entries learned for a given MAC address, interface or VLAN. Use this command without options to clear any learned entries.

Use the optional `instance` parameter to clear the filtering database entries associated with a specified MSTP instance Note that you must first specify a switch port interface before you can specify an MSTP instance.

Compare this usage and operation with the **clear mac address-table static** command on page 17.4. Note that an MSTP instance cannot be specified with **clear mac address-table static**.

Examples This example shows how to clear all dynamically learned filtering database entries for all interfaces, addresses, VLANs.

```
awplus# clear mac address-table dynamic
```

This example shows how to clear all dynamically learned filtering database entries when learned through switch operation for a given MAC address.

```
awplus# clear mac address-table dynamic address 0202.0202.0202
```

This example shows how to clear all dynamically learned filtering database entries when learned through switch operation for a given MSTP instance `1` on switch port interface `port1.0.2`.

```
awplus# clear mac address-table dynamic interface port1.0.2
        instance 1
```

Related Commands **clear mac address-table static**
show mac address-table

clear port counter

Use this command to clear the packet counters of the port.

Syntax `clear port counter [<port>]`

Parameter	Description
<port>	The port number or range

Mode Privileged Exec

Example To clear the packet counter for `port1.0.1`, use the command:

```
awplus#  clear port counter port1.0.1
```

Related Commands show platform port

debug loopprot

This command enables Loop Protection debugging.

The **no** variant of this command disables Loop Protection debugging.

Syntax

```
debug loopprot {info|msg|pkt|state|nsm|all}
no debug loopprot {info|msg|pkt|state|nsm|all}
```

Parameter	Description
`info`	General Loop Protection information.
`msg`	Received and transmitted Loop Detection Frames (LDFs).
`pkt`	Echo raw ASCII display of received and transmitted LDF packets to the console.
`state`	Loop Protection states transitions.
`nsm`	Network Service Module information.
`all`	All debugging information.

Mode Privileged Exec and Global Configuration

Example To enable debug for all state transitions, use the command:

```
awplus#  debug loopprot state
```

Related Commands **show debugging loopprot**
undebug loopprot

debug platform packet

This command enables platform to CPU level packet debug functionality on the switch.

Use the **no** variant of this command to disable platform to CPU level packet debug. If the result means both send and receive packet debug are disabled, then any active timeout will be canceled.

Syntax

```
debug platform packet [recv] [send] [sflow] [timeout <timeout>]
    [vlan <vlan-id>|all]

no debug platform packet [recv] [send]
```

Parameter	Description
`recv`	Debug packets received.
`send`	Debug packets sent.
`sflow`	Debug sFlow packets.
`timeout`	Stop debug after a specified time.
`<timeout>`	`<0-3600>`The timeout period, specified in seconds.
`vlan`	Limit debug to a single VLAN ID specified.
`<vlan-id>`	`<1-4094>` The VLAN ID to limit the debug output on.
`all`	Debug all VLANs (default setting).

Default A 5 minute timeout is configured by default if no other timeout duration is specified.

Mode Privileged Exec and Global Configuration

Usage This command can be used to trace packets sent and received by the CPU. If a timeout is not specified, then a default 5 minute timeout will be applied.

If a timeout of 0 is specified, packet debug will be generated until the **no** variant of this command is used or another timeout value is specified. The timeout value applies to both send and receive debug and is updated whenever the **debug platform packet** command is used.

Examples To enable both receive and send packet debug for the default timeout of 5 minutes, enter:

```
awplus# debug platform packet
```

To enable receive packet debug for 10 seconds, enter:

```
awplus# debug platform packet recv timeout 10
```

To enable packet debug for sFlow packets only for the default timeout of 5 minutes, enter:

```
awplus# debug platform packet sflow
```

To enable send packet debug with no timeout, enter:

```
awplus# debug platform packet send timeout 0
```

To enable VLAN packet debug for VLAN 2 with a timeout duration of 3 minutes, enter:

```
awplus# debug platform packet vlan 2 timeout 150
```

To disable receive packet debug, enter:

```
awplus# no debug platform packet recv
```

Related Commands

show debugging platform packet
undebug platform packet

duplex

This command changes the duplex mode for the specified port.

To see the currently-negotiated duplex mode for ports whose links are up, use the command **show interface**. To see the configured duplex mode (when different from the default), use the command **show running-config**.

Syntax `duplex {auto|full|half}`

Parameter	Description
`auto`	Auto-negotiate duplex mode.
`full`	Operate in full duplex mode only.
`half`	Operate in half duplex mode only.

Default By default, ports auto-negotiate duplex mode (except for 100Base-FX ports which do not support auto-negotiation, so default to full duplex mode).

Mode Interface Configuration

Usage Switch ports in a static or dynamic (LACP) channel group must have the same port speed and be in full duplex mode. Once switch ports have been aggregated into a channel group, you can set the duplex mode of all the switch ports in the channel group by applying this command to the channel group.

Examples To specify full duplex for `port1.0.4`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# duplex full
```

To specify half duplex for `port1.0.4`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# duplex half
```

To auto-negotiate duplex mode for `port1.0.4`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# duplex auto
```

Related Commands
backpressure
polarity
speed
show interface

flowcontrol (switch port)

Use this command to enable flow control, and configure the flow control mode for the switch port.

Use the **no** variant of this command to disable flow control for the specified switch port.

Syntax

```
flowcontrol {send|receive} {off|on}
no flowcontrol
```

Parameter	Description
`receive`	When the port receives pause frames, it temporarily stops (pauses) sending traffic.
`on`	Enable the specified flow control.
`off`	Disable the specified flow control.
`send`	When the port is congested (receiving too much traffic), it sends pause frames to request the other end to temporarily stop (pause) sending traffic.

Default By default, flow control is disabled.

Mode Interface Configuration

Usage The flow control mechanism specified by 802.3x is only for full duplex links. It operates by sending PAUSE frames to the link partner to temporarily suspend transmission on the link

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion, and cannot receive any more traffic, it notifies the other port to stop sending until the condition clears. When the local device detects congestion at its end, it notifies the remote device by sending a pause frame. On receiving a pause frame, the remote device stops sending data packets, which prevents loss of data packets during the congestion period.

Flow control is not recommended when running QoS or ACLs, because the complex queuing, scheduling, and filtering configured by QoS or ACLs may be slowed by applying flow control.

For half-duplex links, an older form of flow control known as backpressure is supported. See the related **backpressure** command on page 17.2.

For flow control on async serial (console) ports, see the **flowcontrol hardware (asyn/console)** command on page 5.11.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# flowcontrol receive on
```

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# flowcontrol receive off
```

Validation Commands show running-config

Related Commands backpressure

linkflap action

Use this command to detect flapping on all ports. If more than 15 flaps occur in less than 15 seconds the flapping port will shut down.

Use the **no** variant of this command to disable flapping detection at this rate.

Syntax

```
linkflap action [shutdown]
no linkflap action
```

Parameter	Description
`linkflap`	Global setting for link flapping.
`action`	Specify the action for port.
`shutdown`	Shutdown the port.

Default Linkflap action is disabled by default.

Mode Global Configuration

Example To enable the linkflap action command on the switch, use the following commands:

```
awplus# configure terminal
awplus(config)# linkflap action shutdown
```

loop-protection

Use this command to enable the loop-protection loop-detection feature, and configure the detection mechanism parameters.

Use the **no** variant of this command to disable the loop-protection loop-detection feature.

Syntax

```
loop-protection loop-detect [ldf-interval <period>]
    [ldf-rx-window <frames>]

no loop-protection [loop-detect]
```

Parameter	Description
`loop-detect`	Enables loop detection when used with loop-protection keywords. Disables loop detection when used with no loop-protection keywords.
`ldf-interval`	The time (in seconds) between successive loop-detect frames being sent.
<period>	Specify a period between 1 and 600 seconds. The default is 10 seconds.
`ldf-rx-window`	The number of transmitted loop detection frames whose details are held for comparing with frames arriving at the same port.
<frames>	Specify a value for the window size between 1 and 5 frames. The default is 3 frames.

Default The loop-protection loop-detection feature is disabled by default. The default interval is 10 seconds, and the default window size is 3 frames.

Mode Global Configuration

Usage See the **Loop Protection** section in **Chapter 16, Switching Introduction** for relevant conceptual, configuration, and overview information prior to applying this command.

Example To enable the loop-detect mechanism on the switch, and generate loop-detect frames once every 5 seconds, use the following commands:

```
awplus# configure terminal
awplus(config)# loop-protection loop-detect ldf-interval 5
```

Related Commands

loop-protection action
loop-protection timeout
show loop-protection
thrash-limiting

loop-protection action

Use this command to specify the protective action to apply when a network loop is detected on an interface.

Use the **no** variant of this command to reset the loop protection actions to the default action, vlan-disable, on an interface.

Syntax

```
loop-protection action {link-down|log-only|port-disable|vlan-disable|
   none}
no loop-protection action
```

Parameter	Description
`link-down`	Block all traffic on a port (or aggregated link) that detected the loop, and take **down** the link.
`log-only`	Details of loop conditions are logged. No action is applied to the port (or aggregated link).
`port-disable`	Block all traffic on interface for which the loop occurred, but keep the link in the **up** state.
`vlan-disable`	Block all traffic for the VLAN on which the loop traffic was detected. Note that setting this parameter will also enable ingress filtering. This is the default action.
`none`	Applies no protective action.

Default loop-protection action vlan-disable

Mode Interface Configuration

Usage See the **Loop Protection** section in **Chapter 16, Switching Introduction** for relevant conceptual, configuration, and overview information prior to applying this command.

Example To disable an interface (`port1.0.4`), and bring the link down, when a network loop is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# loop-protection action link-down
```

Related Commands

loop-protection
loop-protection timeout
show loop-protection
thrash-limiting

loop-protection timeout

Use this command to specify the Loop Protection recovery action duration on an interface.

Use the **no** variant of this command to set the loop protection timeout to the default.

Syntax

```
loop-protection timeout <duration>
no loop-protection timeout
```

Parameter	Description
<duration>	The time (in seconds) for which the configured action will apply before being disabled. This duration can be set between 0 and 86400 seconds (24 hours). The set of 0 means infinity so timeout does not expire.

Default The default is 7 seconds.

Mode Interface Configuration

Usage See the **Loop Protection** section in **Chapter 16, Switching Introduction** for relevant conceptual, configuration, and overview information prior to applying this command.

Example To configure a loop protection action timeout of 10 seconds for `port1.0.4`, use the command:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# loop-protection timeout 10
```

Related Commands

loop-protection
loop-protection action
show loop-protection
thrash-limiting

mac address-table acquire

Use this command to enable MAC address learning on the device.

Use the **no** variant of this command to disable learning.

Syntax

```
mac address-table acquire
no mac address-table acquire
```

Default Learning is enabled by default for all instances.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# mac address-table acquire
```

mac address-table ageing-time

Use this command to specify an ageing-out time for a learned MAC address. The learned MAC address will persist for at least the specified time.

The **no** variant of this command will reset the ageing-out time back to the default of 300 seconds (5 minutes).

Syntax

```
mac address-table ageing-time <ageing-timer> none
no mac address-table ageing-time
```

Parameter	Description
<ageing-timer>	*<10-1000000>* The number of seconds of persistence.
none	Disable learned MAC address timeout.

Default The default ageing time is 300 seconds.

Mode Global Configuration

Examples The following commands specify various ageing timeouts on the switch:

```
awplus# configure terminal
awplus(config)# mac address-table ageing-time 1000

awplus# configure terminal
awplus(config)# mac address-table ageing-time none

awplus# configure terminal
awplus(config)# no mac address-table ageing-time
```

mac address-table static

Use this command to statically configure the MAC address-table to forward or discard frames with a matching destination MAC address.

Syntax

```
mac address-table static <mac-addr> {forward|discard} interface
   <port> [vlan <vid>]

no mac address-table static <mac-addr> {forward|discard} interface
   <port> [vlan <vid>]
```

Parameter	Description
<mac-addr>	The destination MAC address in HHHH.HHHH.HHHH format.
<port>	The port to display information about. The port may be a switch port (e.g. port1.0.4), a static channel group (e.g. sa2), or a dynamic (LACP) channel group (e.g. po2).
<vid>	The VLAN ID. If you do not specify a VLAN, its value defaults to vlan 1.

Mode Global Configuration

Usage The **mac address-table static** command is only applicable to Layer 2 switched traffic within a single VLAN. Do not apply the **mac address-table static** command to Layer 3 switched traffic passing from one VLAN to another VLAN. Frames will not be discarded across VLANs because packets are routed across VLANs. This command only works on Layer 2 traffic.

Example

```
awplus# configure terminal

awplus(config)# mac address-table static 2222.2222.2222 forward
                interface port1.0.4 vlan 3
```

Related Commands clear mac address-table static
show mac address-table

mac address-table thrash-limit

Use this command to set the thrash limit on the switch or stack. Thrashing occurs when a MAC address table rapidly "flips" its mapping of a single MAC address between two subnets, usually as a result of a network loop.

Use the **no** variant of this command to disable thrash limiting.

Syntax

```
mac address-table thrash-limit <rate>
no mac address-table thrash-limit
```

Parameter	Description
<rate>	sets the maximum thrash rate at which limiting is applied. This rate can be set between 5 and 255 MAC thrashing flips per second. Once the thrash limit rate is reached, the port is considered to be thrashing.

Default No thrash limiting

Mode Global Configuration

Usage Use this command to limit thrashing on the selected port range.

Example To apply a thrash limit of 100 MAC address flips per second:

```
awplus# configure terminal
awplus(config)# mac address-table thrash-limit 100
```

Related Commands **show mac address-table thrash-limit**

mirror interface

Use this command to define a mirror port and mirrored (monitored) ports and direction of traffic to be mirrored. The port for which you enter interface mode will be the mirror port.

The destination port is removed from all VLANs, and no longer participates in other switching.

Use the **no** variant of this command to disable port mirroring by the destination port on the specified source port.

Use the **none** variant of this command when using copy-to-mirror ACL and QoS commands.

Syntax

```
mirror interface <source-port-list> direction {both|receive|transmit}
mirror interface none
no mirror interface <source-port-list>
no mirror interface none
```

Parameter	Description
<source-port-list>	The source switch ports to mirror. A port-list can be: ■ a port (e.g. `port1.0.2`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.2` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.0.6` The source port list cannot include dynamic or static channel groups (link aggregators).
`direction`	Specifies whether to mirror traffic that the source port receives, transmits, or both.
`both`	Mirroring traffic both received and transmitted by the source port.
`receive`	Mirroring traffic received by the source port.
`transmit`	Mirroring traffic transmitted by the source port.
`none`	Specify this parameter for use with the ACL (Access Control List) **access-list** and QoS (Quality of Service) default action commands when used with the **copy-to-mirror** parameter option, so you can specify the destination port (the analyzer port) for the traffic without specifying a source mirror port. See the ACL commands **access-list (hardware IP numbered)** and **access-list (hardware MAC numbered)**, and the QoS command **default-action** for further information.

Mode Interface Configuration

Usage Use this command to send traffic to another device connected to the mirror port for monitoring.

See **"Port Mirroring" on page 16.17**.

A mirror port cannot be associated with a VLAN. If a switch port is configured to be a mirror port, it is automatically removed from any VLAN it was associated with.

This command can only be applied to a single mirror (destination) port, not to a range of ports, nor to a static or dynamic channel group. Do not apply multiple interfaces with an interface command before issuing the mirror interface command. One interface may have multiple mirror interfaces.

Example To mirror traffic received and transmitted on `port1.0.4` and `port1.0.5` to destination `port1.0.3`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# mirror interface port1.0.4,port1.0.5
                   direction both
```

To enable use with the **access-list (hardware IP numbered)** ACL and **default-action** QoS commands to destination `port1.0.3` without specifying a source port, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# mirror interface none
```

To mirror all TCP traffic, received or transmitted to analyzer `port1.0.1`, see the sample config below:

```
awplus#show running-config

!
mls qos enable
access-list 3000 copy-to-mirror tcp any any
access-group 3000
!
interface port1.0.1
 mirror interface none
 switchport
!
```

Related Commands
access-list (hardware IP numbered)
access-list (hardware MAC numbered)
default-action

platform hwfilter-size

You can use this command to control the configuration of hardware Access Control Lists (ACLs), which determines the total available number and functionality of hardware ACLs.

For this command to take effect, you need to reboot the affected service.

One cannot attach an IPv6 ACL to a port if the ACL contains a specified source or destination IPv6 address or both and the **hw-filter size** setting is **ipv4-limited-ipv6**. If you do so, a diagnostic message will be generated.

Syntax `platform hwfilter-size {ipv4-limited-ipv6|ipv4-full-ipv6}`

Parameter	Description
`hwfilter-size`	Configure hardware ACLs command.
`ipv4-full-ipv6`	Configure hardware ACLs to filter IPv4 traffic, MAC addresses and IPv6 traffic, including filtering on source or destination IPv6 addresses, or both; however, this will reduce the total number of filters available in the hardware table.
`ipv4-limited-ipv6`	Configure hardware ACLs to filter IPv4 traffic, MAC addresses and IPv6 traffic. Source or destination IPv6 addresses or both are not filtered.

Default The default mode is **ipv4-limited-ipv6**.

Mode Global Configuration

Example To configure hardware ACLs to filter IPv4 and IPv6 traffic, use the following commands:

```
awplus# configure terminal
awplus(config)# platform hwfilter-size ipv4-full-ipv6
```

Related Commands **show platform**
ipv6 access-list (named)

platform load-balancing

This command selects which address fields are used as inputs into the load balancing algorithm for aggregated links. The output from this algorithm is used to select which individual path a given packet will traverse within an aggregated link.

The **no** variant of this command applies its default setting.

Syntax

```
platform load-balancing {src-dst-mac|src-dst-ip}
no platform load-balancing
```

Parameter	Description
src-dst-mac	Include the source and destination MAC addresses (Layer 2)
src-dst-ip	Include the source and destination IP addresses (Layer 3) and UDP/TCP source and destination ports. If you choose this option, the algorithm will use MAC addresses to calculate load balancing for Layer 2 and non-IP packets.

Default The default is **src-dst-ip**.

Mode Global configuration

Examples To set the load balancing algorithm to include only Layer 2 MAC addresses, enter:

```
awplus# configure terminal
awplus(config)# platform load-balancing src-dst-mac
```

To set the load balancing algorithm to include only Layer 3 IP addresses and L4 ports, enter:

```
awplus# configure terminal
awplus(config)# platform load-balancing src-dst-ip
```

Related Commands **show platform**

platform stop-unreg-mc-flooding

This command stops multicast packets flooding out of all the ports in the VLAN until these packets are registered. This command does this by sending unregistered multicast packets to the switch processor, so there is no flooding of the multicast traffic onto the VLAN. Unregistered traffic will not flow until the switch has registered it, regardless of attempts to subscribe to it. Once the traffic is registered, it flows to registered subscribers and ports.

Use the **no** variant of this command to revert to default behavior and disable this feature.

Note

This command should not be used within any IPv6 networks.
IPv6 neighbor discovery operation is inhibited by this feature.
This command does not stop reserved Local Network Control Block IPv4 multicast packets in the address range 224.0.0.1 to 224.0.0.255 (224.0.0/24).
See **www.iana.org/assignments/multicast-addresses/multicast-addresses.xml#multicast-addresses-1**

Syntax

```
platform stop-unreg-mc-flooding

no platform stop-unreg-mc-flooding
```

Default This feature is disabled by default.

Mode Global Configuration

Usage This command stops the periodic flooding of unknown or unregistered multicast packets when the Group Membership interval timer expires and there are no subscribers to a multicast group. If there is multicast traffic in a VLAN without subscribers, multicast traffic temporarily floods out of the VLAN when the Group Membership interval timer expires, which happens when the switch does not get replies from Group Membership queries.

For further information about multicast groups see the **Multicast Groups** section in **Chapter 47, Multicast Introduction and Commands**.

For further information about query messages see the **Staying in the Multicast Group (Query Message)** section in **Chapter 48, IGMP and IGMP Snooping Introduction**.

This command also stops the initial flood of multicast packets that happens when a new multicast source starts to send traffic. This flooding lasts until snooping recognizes the multicast group. For example, in sites where IP cameras have multicast groups, traffic is flooded to the VLAN and causes large bursts of traffic. Use this command when there is limited processing available for large bursts of traffic, such as in sites with IP cameras.

Output See the console message warning about IPv6 operation after entering this command:

```
% WARNING: IPv6 will not work with this setting enabled
% Please consult the documentation for more information
```

See these sample console messages when the Group Membership interval timer expires, which happens when the switch does not get replies from Group Membership queries:

```
awplus: [MLD-EVENTS] Grp - Rec Liveness Timer: Expiry for Grp ff0e::1 on port1.2.7
awplus: [IGMP-EVENTS] : Expiry (Unreg MC Timer) for Grp 224.2.2.2 on vlan4
```

Examples To enable this feature and stop multicast packet flooding, use the following commands:

```
awplus# configure terminal
awplus(config)# platform stop-unreg-mc-flooding
```

To disable this feature and allow multicast packet flooding, use the following commands:

```
awplus# configure terminal
awplus(config)# no platform stop-unreg-mc-flooding
```

Related Commands **show platform**
show running-config

platform vlan-stacking-tpid

This command specifies the Tag Protocol Identifier (TPID) value that applies to all frames that are carrying double tagged VLANs. All nested VLANs must use the same TPID value. (This feature is sometimes referred to as VLAN stacking or VLAN double-tagging.)

Use the **no** variant of this command to revert to the default TPID value (0x8100).

Note

Because the additional tag increases the frame size beyond 1522 bytes, you must increase the MRU size to activate VLAN-stacking. Go into interface mode for the appropriate ports and use the **mru** command on page 14.5.

Syntax

```
platform vlan-stacking-tpid <tpid>
no platform vlan-stacking-tpid
```

Parameter	Description
<tpid>	The Ethernet type of the tagged packet, as a two byte hexadecimal number.

Default The default TPID value of 0x8100 is restored using a **no platform vlan-stacking-tpid** command.

Mode Global Configuration

Examples To set the VLAN stacking TPID value to 0x9100, use the following commands:

```
awplus# configure terminal
awplus(config)# platform vlan-stacking-tpid 9100
```

To reset the VLAN stacking TPID value to the default (0x8100), use the following commands:

```
awplus# configure terminal
awplus(config)# no platform vlan-stacking-tpid
```

Related Commands

switchport vlan-stacking (double tagging)
show platform
show running-config

polarity

This command sets the MDI/MDIX polarity on a copper-based switch port.

Syntax `polarity {auto|mdi|mdix}`

Parameter	Description
`mdi`	Sets the polarity to MDI (medium dependent interface).
`mdix`	Sets the polarity to MDI-X (medium dependent interface crossover).
`auto`	The switch port sets the polarity automatically. This is the default option.

Default By default, switch ports set the polarity automatically (**auto**).

Mode Interface Configuration

Usage We recommend the default **auto** setting for MDI/MDIX polarity. Polarity applies to copper 10BASE-T, 100BASE-T, and 1000BASE-T switch ports; It does not apply to fiber ports. For more information, see **"MDI/MDIX Connection Modes" on page 16.5**.

Example To set the polarity for `port1.0.6` to fixed MDI mode, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# polarity mdi
```

show debugging loopprot

This command shows Loop Protection debugging information.

Syntax `show debugging loopprot`

Mode User Exec and Privileged Exec

Example To display the enabled Loop Protection debugging modes, use the command:

```
awplus# show debugging loopprot
```

Related Commands **debug loopprot**

show debugging platform packet

This command shows platform to CPU level packet debugging information.

Syntax `show debugging platform packet`

Mode User Exec and Privileged Exec

Example To display the platform packet debugging information, use the command:

```
awplus# show debugging platform packet
```

Related Commands **debug platform packet**
undebug platform packet

show flowcontrol interface

Use this command to display flow control information.

Syntax `show flowcontrol interface <port>`

Parameter	Description
<port>	Specifies the name of the port to be displayed.

Mode User Exec and Privileged Exec

Example To display the flow control for the `port1.0.5`, use the command:

```
awplus# show flowcontrol interface port1.0.5
```

Output **Figure 17-1: Example output from the show flowcontrol interface command for a specific interface**

```
Port     Send  FlowControl    Receive  FlowControl  RxPause TxPause
          admin    oper        admin    oper
-----     ------- --------    ------- --------     ------- -------
port1.0.5 on       on           on       on           0       0
```

show interface switchport

Use this command to show VLAN information about each switch port.

Syntax `show interface switchport`

Mode User Exec and Privileged Exec

Example To display VLAN information about each switch port, enter the command:

```
awplus# show interface switchport
```

Output **Figure 17-2: Example output from the show interface switchport command**

```
 Interface name          : port1.0.1
 Switchport mode         : access
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            :  2
 Configured Vlans        : 2

 Interface name          : port1.0.2
 Switchport mode         : trunk
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            :  1
 Configured Vlans        : 1 4 5 6 7 8
...
```

Related Commands show interface memory

show loop-protection

Use this command to display the current loop protection setup for the device.

Syntax `show loop-protection [interface <port-list>] [counters]`

Parameter	Description
`interface`	The interface selected for display.
`<port-list>`	A port, a port range, or an aggregated link.
`counters`	Displays counter information for loop protection.

Mode User Exec and Privileged Exec

Usage This command is used to display the current configuration and operation of the Loop Protection feature

Examples To display the current configuration status for `port1.0.1`, use the command:

```
awplus# show loop-protection interface port1.0.1
```

Figure 17-3: Example output from the show loop-protection command

```
Loop-Detection:          Enabled
LDF Interval:            10 [sec]
Interface:               port1.0.1
 Action:                 port-disable
 Timeout:                300 [sec]
 Vlan:                   1
  Status:                Blocking
  Timeout Remaining:     115 [sec]
 Vlan:                   2
  Status:                Normal
  Timeout Remaining:     0 [sec]
```

To display the counter information for `port1.0.1`, use the command:

```
awplus# show loop-protection interface port1.0.1 counters
```

Figure 17-4: Example output from the show loop-protection interface counters command for port1.0.1

```
Interface:               port1.0.1
 Vlan:                   1
  LDF Tx:                3
  LDF Rx:                1
  Invalid LDF Rx:        1
  Action:                1
Vlan:                    2
  LDF Tx:                3
  LDF Rx:                0
  Invalid LDF Rx:        0
  Action:                0
```

show mac address-table

Use this command to display the mac address-table for all configured VLANs.

Syntax `show mac address-table`

Mode User Exec and Privileged Exec

Usage The **show mac address-table** command is only applicable to view a mac address-table for Layer 2 switched traffic within VLANs.

Example To display the mac address-table, use the following command:

```
awplus# show mac address-table
```

Output See the below sample output captured when there was no traffic being switched:

```
awplus#show mac address-table

VLAN Port          MAC               State
1    unknown       0000.cd28.0752    static
ARP  -             0000.cd00.0000    static
```

See the sample output captured when packets were switched and mac addresses were learned:

```
awplus#show mac address-table

VLAN Port          MAC               State
1    unknown       0000.cd28.0752    static
1    port1.0.6     0030.846e.9bf4    dynamic
1    port1.0.4     0030.846e.bac7    dynamic
ARP  -             0000.cd00.0000    static
```

Note the new mac addresses learned for `port1.0.4` and `port1.0.6` added as dynamic entries.

Note the first column of the output below shows VLAN IDs if multiple VLANs are configured:

```
awplus#show mac address-table

VLAN Port          MAC               State
1    unknown       0000.cd28.0752    static
1    port1.0.4     0030.846e.bac7    dynamic
2    unknown       0000.cd28.0752    static
2    port1.0.6     0030.846e.9bf4    dynamic
ARP  -             0000.cd00.0000    static
```

Also note manually configured static mac-addresses are shown to the right of the type

column:

```
awplus(config)#mac address-table static 0000.1111.2222 for int
port1.0.3 vlan 2
awplus(config)#end
awplus#
awplus#show mac address-table

VLAN Port          MAC               State
1    unknown       0000.cd28.0752    static
1    port1.0.2     0030.846e.bac7    dynamic
2    port1.0.3     0000.1111.2222    static
2    unknown       0000.cd28.0752    static
2    port1.0.5     0030.846e.9bf4    dynamic
ARP  -             0000.cd00.0000    statics
```

Related Commands

clear mac address-table dynamic
clear mac address-table static
mac address-table static

show mac address-table thrash-limit

Use this command to display the current thrash limit set for all interfaces on the device.

Syntax

```
show mac address-table thrash-limit
```

Mode

User Exec and Privileged Exec

Example

To display the current, use the following command:

```
awplus# show mac address-table thrash-limit
```

Output

Figure 17-5: Example output from the show mac address-table thrash-limit command

```
% Thrash-limit  7 movements per second
```

Related Commands

mac address-table thrash-limit

show mirror

Use this command to display the status of all mirrored ports.

Syntax `show mirror`

Mode User Exec and Privileged Exec

Example To display the status of all mirrored ports, use the following command:

```
awplus# show mirror
```

Output **Figure 17-6: Example output from the show mirror command**

```
Mirror Test Port Name: port1.0.1
Mirror option: Enabled
Mirror direction: both
Monitored Port Name: port1.0.2
Mirror Test Port Name: port1.0.3
Mirror option: Enabled
Mirror direction: receive
Monitored Port Name: port1.0.4
Mirror Test Port Name: port1.0.3
Mirror option: Enabled
Mirror direction: receive
Monitored Port Name: port1.0.1
Mirror Test Port Name: port1.0.1
Mirror option: Enabled
Mirror direction: receive
Monitored Port Name: port1.0.3
Mirror Test Port Name: port1.0.1
Mirror option: Enabled
Mirror direction: transmit
Monitored Port Name: port1.0.4
```

show mirror interface

Use this command to display port mirroring configuration for a mirrored (monitored) switch port.

Syntax `show mirror interface <port>`

Parameter	Description
<port>	The monitored switch port to display information about.

Mode User Exec, Privileged Exec and Interface Configuration

Example To display port mirroring configuration for the `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# show mirror interface port1.0.4
```

Output **Figure 17-7: Example output from the show mirror interface command**

```
Mirror Test Port Name: port1.0.3
Mirror option: Enabled
Mirror direction: both
Monitored Port Name: port1.0.4
```

show platform

This command displays the settings configured by using the **platform** commands.

Syntax `show platform`

Mode Privileged Exec

Usage This command displays the settings in the running config. For changes in some of these settings to take effect, the switch must be rebooted with the new settings in the startup config.

Example To check the settings configured with **platform** commands on the switch, use the following command:

```
awplus# show platform
```

Output **Figure 17-8: Example output from the show platform command**

```
awplus# show platform
Vlan-stacking TPID           0x8100
```

Table 17-1: Parameters in the output of the show platform command

Parameter	Description
Vlan-stacking TPID	The value of the TPID set in the Ethernet type field when a frame has a double VLAN tag (**platform vlan-stacking-tpid** command on page 17.28).

Related Commands **platform load-balancing**
platform vlan-stacking-tpid

show platform classifier statistics utilization brief

This command displays the number of used entries available for various platform functions, and the percentage that number of entries represents of the total available.

Syntax `show platform classifier statistics utilization brief`

Mode Privileged Exec

Example To display the platform classifier utilization statistics, use the following command:

```
awplus# show platform classifier statistics utilization brief
```

Output **Figure 17-9: Output from the show platform classifier statistics utilization brief command**

```
[Instance 3.0]
(Port1.0.1-1.0.24)
  Number of Entries:
  Policy Type      Group ID   Used / Total
  ---------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS                      -1    0 /    0 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            0 /  768 (  0%)

[Instance 3.1]
(Port1.0.25-1.0.48)
  Number of Entries:
  Policy Type      Group ID   Used / Total
  ---------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS                      -1    0 /    0 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            2 /  768 (  0%)
```

Figure 17-10: Output from the show platform classifier statistics utilization brief command, with the DOS detection feature enabled

```
[Instance 3.0]
[Port1.0.1-1.0.24]
Number of Entries:
  Policy Type      Group ID   Used / Total
  ---------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS              1476395011    0 /  128 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            0 /  640 (  0%)

[Instance 3.1]
[Port1.0.25-1.0.48]
  Number of Entries:
  Policy Type      Group ID   Used / Total
  ---------------------------------------------
  ACL              1476395009    0 /  122 (  0%)
  DoS              1476395011    1 /  128 (  0%)
  VLAN Counter             -1    0 /    0 (  0%)
  QoS                            2 /  640 (  0%)
                            1    2 /  128 (  1%)
```

Related Commands show platform

show platform port

This command displays the various port registers or platform counters for specified switchports.

Syntax `show platform port [<port-list>|counters]`

Parameter	Description
<port-list>	The ports to display information about. A port-list can be: ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.6` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.0.6`.
`counters`	Show the platform counters.

Mode Privileged Exec

Examples To display port registers for `port1.0.1` and `port1.0.2` use the following command:

```
awplus# show platform port port1.0.1-port1.0.2
```

To display platform counters for `port1.0.1` and `port1.0.2` use the following command:

```
awplus# show platform port port1.0.1-port1.0.2 counters
```

Output **Figure 17-11: Example output from the show platform port command**

```
awplus#show platform port port1.0.1
Phy register value for port1.0.1 (ifindex: 5001)

00:1140  01:7949  02:0020  03:60B1  04:01E1  05:0000  06:0004  07:2001
08:0000  09:0600  10:0000  11:0000  12:0000  13:0000  14:0000  15:0000
16:0000  17:0000  18:0000  19:0000  20:0000  21:0000  22:0000  23:0000
24:0000  25:0000  26:0000  27:0000  28:0000  29:0000  30:0000  31:0000

Port configuration for lport 0x08001000:
  enabled:                1
  loopback:               0
  link:                   0
  speed:                  0  max speed:               1000
  duplex:                 0
  linkscan:               2
  autonegotiate:          1
  master:                 2
  tx pause:               1  rx pause:                  1
  untagged vlan:          1
  vlan filter:            3
  stp state:              1
  learn:                  5
  discard:                0
  max frame size:      1522
  MC Disable SA:         no
  MC Disable TTL:        no
  MC egress untag:        0
  MC egress vid:          0
  MC TTL threshold:      -1
```

Table 17-2: Parameters in the output from the show platform port command

Parameter	Description
Ethernet MAC counters	
`Combined receive/ transmit packets by size (octets) counters`	Number of packets in each size range received and transmitted.
`64`	Number of 64 octet packets received and transmitted.
`65 - 127`	Number of 65 - 127 octet packets received and transmitted.
`128 - 255`	Number of 128 - 255 octet packets received and transmitted.
`256 - 511`	Number of 256 - 511 octet packets received and transmitted.
`512 - 1023`	Number of 512 - 1023 octet packets received and transmitted.
`1024 - MaxPktSz`	Number of packets received and transmitted with size 1024 octets to the maximum packet length.
`1519 - 1522`	Number of 1519 - 1522 octet packets received and transmitted.
`1519 - 2047`	Number of 1519 - 2047 octet packets received and transmitted.
`2048 - 4095`	Number of 2048 - 4095 octet packets received and transmitted.
`4096 - 9216`	Number of 4096 - 9216 octet packets received and transmitted.
General Counters	
`Receive`	Counters for traffic received.
`Octets`	Number of octets received.
`Pkts`	Number of packets received.
`FCSErrors`	Number of FCS (Frame Check Sequence) error events received.
`UnicastPkts`	Number of unicast packets received.
`MulticastPkts`	Number of multicast packets received.
`BroadcastPkts`	Number of broadcast packets received.
`PauseMACCtlFrms`	Number of Pause MAC Control Frames received.
`OversizePkts`	Number of oversize packets received.
`Fragments`	Number of fragments received.
`Jabbers`	Number of jabber frames received.
`UnsupportOpcode`	Number of MAC Control frames with unsupported opcode received.
`AlignmentErrors`	Receive Alignment Error Frame Counter.
`SysErDurCarrier`	Receive Code Error Counter.

Table 17-2: Parameters in the output from the show platform port command (cont.)

Parameter	Description
CarrierSenseErr	Receive False Carrier Counter.
UndersizePkts	Number of undersized packets received.
Transmit	Counters for traffic transmitted.
Octets	Number of octets transmitted.
Pkts	Number of packets transmitted.
UnicastPkts	Number of unicast packets transmitted.
MulticastPkts	Number of multicast packets transmitted.
BroadcastPkts	Number of broadcast packets transmitted.
PauseMACCtlFrms	Number of Pause MAC Control Frames transmitted.
OversizePkts	Number of oversize packets transmitted.
FrameWDeferrdTx	Transmit Single Deferral Frame counter.
FrmWExcesDefer	Transmit Multiple Deferral Frame counter.
SingleCollsnFrm	Transmit Single Collision Frame counter.
MultCollsnFrm	Transmit Multiple Collision Frame counter.
LateCollisions	Transmit Late Collision Frame counter.
ExcessivCollsns	Transmit Excessive Collision Frame counter.
Collisions	Transmit Total Collision counter
Layer 3 Counters	
ifInUcastPkts	Inbound interface Unicast counter.
ifInDiscards	Inbound interface Discarded Packets counter.
ipInHdrErrors	Inbound interface Header Errors counter.
ifOutUcastPkts	Outbound interface Unicast counter.
ifOutErrors	Outbound interface Error counter.
Miscellaneous Counters	
DropEvents	Drop Event counter
ifOutDiscards	Outbound interface Discarded Packets counter.
MTUExcdDiscard	Receive MTU Check Error Frame Counter

Output **Figure 17-12: Example output from the show platform port counters command**

```
awplus#show platform port port1.0.1 counters

 Switch Port Counters
------------------------------------------------------------------------

Port port1.0.1 Ethernet MAC counters:
 Combined receive/transmit packets by size (octets) counters:
  64                              0 1024 - MaxPktSz                  0
  65 - 127                        0 1519 - 1522                      0
  128 - 255                       0 1519 - 2047                      0
  256 - 511                       0 2048 - 4095                      0
  512 - 1023                      0 4096 - 9216                      0

 General Counters:
 Receive                            Transmit
  Octets                          0 Octets                           0
  Pkts                            0 Pkts                             0
  FCSErrors                       0
  UnicastPkts                     0 UnicastPkts                      0
  MulticastPkts                   0 MulticastPkts                    0
  BroadcastPkts                   0 BroadcastPkts                    0
  PauseMACCtlFrms                 0 PauseMACCtlFrms                  0
  OversizePkts                    0
  Fragments                       0
  Jabbers                         0
  UnsupportOpcode                 0
  AlignmentErrors                 0
  SymErDurCarrier                 0
  CarrierSenseErr                 0
  UndersizePkts                   0
                                     FrameWDeferrdTx                 0
                                     FrmWExcesDefer                  0
                                     SingleCollsnFrm                 0
                                     MultCollsnFrm                   0
                                     LateCollisions                  0
                                     ExcessivCollsns                 0
                                     Collisions                      0

 Layer 3 Counters:
  ifInUcastPkts                   0 ifOutUcastPkts                   0
  ifInDiscards                    0 ifOutErrors                      0
  ipInHdrErrors                   0

 Miscellaneous Counters:
  DropEvents                      0
  ifOutDiscards                   0
  MTUExcdDiscard                  0

------------------------------------------------------------------------
```

Table 17-3: Output parameters from the show platform port counters command

Parameter	Description
Ethernet MAC counters	
`Combined receive/transmit packets by size (octets) counters`	Number of packets in each size range received and transmitted.
`64`	Number of 64 octet packets received and transmitted.
`65 - 127`	Number of 65 - 127 octet packets received and transmitted.

Table 17-3: Output parameters from the show platform port counters command

Parameter	Description
128 - 255	Number of 128 - 255 octet packets received and transmitted.
256 - 511	Number of 256 - 511 octet packets received and transmitted.
512 - 1023	Number of 512 - 1023 octet packets received and transmitted.
1024 - MaxPktSz	Number of packets received and transmitted with size 1024 octets to the maximum packet length.
General Counters	
Receive	Counters for traffic received.
Octets	Number of octets received.
Pkts	Number of packets received.
CRCErrors	Number of CRC (Cyclic Redundancy Check) error events received.
UnicastPkts	Number of unicast packets received.
MulticastPkts	Number of multicast packets received.
BroadcastPkts	Number of broadcast packets received.
FlowCtrlFrms	Number of good Flow Control frames received.
OversizePkts	Number of oversize packets received.
Fragments	Number of fragments received.
Jabbers	Number of jabber frames received.
UnsupportOpcode	Number of MAC Control frames with unsupported opcode received.
UndersizePkts	Number of undersized packets received.
Transmit	Counters for traffic transmitted.
Octets	Number of octets transmitted.
Pkts	Number of packets transmitted.
UnicastPkts	Number of unicast packets transmitted.
MulticastPkts	Number of multicast packets transmitted.
BroadcastPkts	Number of broadcast packets transmitted.
FlowCtrlFrms	Number of good Flow Control frames transmitted.
OversizePkts	Number of oversize packets transmitted.
FlowCtrlFrms	The number of Flow Control frames transmitted.
Collisions	Total number of collisions seen by the MAC.
LateCollisions	Total number of late collisions seen by the MAC.
ExcessivCollsns	Number of frames dropped in the transmit MAC due to excessive collisions. This is applicable for Half-Duplex mode only.

Table 17-3: Output parameters from the show platform port counters command

Parameter	Description
Miscellaneous Counters	
`Mac TxErr`	Number of frames not transmitted correctly or dropped due to internal MAC transmit error.
`Mac RxErr`	Number of Receive Error events seen by the receive side of the MAC.
`DropEvents`	Number of instances that the port was unable to receive packets due to insufficient bandwidth to one of the PP internal resources, such as the DRAM or buffer allocation.

show port-security interface

Use this command to show the current port-security configuration and the switch port status.

Syntax `show port-security interface <port>`

Parameter	Description
`<port>`	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Privileged Exec

Example To see the port-security status on `port1.0.1`, use the following command:

```
awplus# show port-security interface port1.0.1
```

Output **Figure 17-13: Example output from the show port-security interface command**

```
Port Security configuration
Security Enabled               : YES
Port Status                    : ENABLED
Violation Mode                 : TRAP
Aging                          : OFF
Maximum MAC Addresses          : 3
Total MAC ddresses             : 1
Lock Status                    : UNLOCKED
Security Violation Count       : 0
Last Violation Source Address : None
```

show port-security intrusion

Shows the intrusion list. If the port is not specified, the entire intrusion table is shown.

Syntax `show port-security intrusion [interface <port>]`

Parameter	Description
`interface`	Specify a port
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Privileged Exec

Example To see the intrusion list on `port1.0.1`, use the following command:

```
awplus# show port-security intrusion interface port1.0.1
```

Output **Figure 17-14: Example output from the show port-security intrusion command for port 1.0.1**

```
Port Security Intrusion List
Interface: port1.0.1 -3 intrusion(s) detected
11-22-33-44-55-04 11-22-33-44-55-06 11-22-33-44-55-08
```

show storm-control

Use this command to display storm-control information for all interfaces or a particular interface.

Syntax `show storm-control [<port>]`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode User Exec and Privileged Exec

Example To display storm-control information for `port1.0.2`, use the following command:

```
awplus# show storm-control port1.0.2
```

Output **Figure 17-15: Example output from the show storm-control command for port1.0.2**

```
Port       BcastLevel  McastLevel  DlfLevel
port1.0.2      40. 0%     100. 0%   100. 0%
```

Example To display storm-control information for all ports, use the following command:

```
awplus# show storm-control
```

Output **Figure 17-16: Example output from the show storm-control command for all ports**

```
awplus#show storm-control
Port       BcastLevel  McastLevel  DlfLevel
port1.0.1      100.0%      100.0%    100.0%
port1.0.2      100.0%      100.0%    100.0%
port1.0.3      100.0%      100.0%    100.0%
port1.0.4      100.0%      100.0%    100.0%
port1.0.5      100.0%      100.0%    100.0%
port1.0.6      100.0%      100.0%    100.0%
port1.0.7      100.0%      100.0%    100.0%
port1.0.8      100.0%      100.0%    100.0%
port1.0.9      100.0%      100.0%    100.0%
port1.0.10     100.0%      100.0%    100.0%
port1.0.11     100.0%      100.0%    100.0%
port1.0.12     100.0%      100.0%    100.0%
port1.0.13     100.0%      100.0%    100.0%
port1.0.14     100.0%      100.0%    100.0%
port1.0.15     100.0%      100.0%    100.0%
port1.0.16     100.0%      100.0%    100.0%
port1.0.17     100.0%      100.0%    100.0%
port1.0.18     100.0%      100.0%    100.0%
port1.0.19     100.0%      100.0%    100.0%
port1.0.20     100.0%      100.0%    100.0%
port1.0.21     100.0%      100.0%    100.0%
port1.0.22     100.0%      100.0%    100.0%
port1.0.23     100.0%      100.0%    100.0%
port1.0.24     100.0%      100.0%    100.0%
```

Related Commands storm-control level

speed

This command changes the speed of the specified port. You can optionally specify the speed or speeds that get autonegotiated, so autonegotiation is only attempted at the specified speeds.

To see the currently-negotiated speed for ports whose links are up, use the **show interface** command. To see the configured speed (when different from the default), use the **show running-config** command.

Syntax `speed {10|100|1000|10000|auto [10][100][1000][10000]}`

The following table shows the speed options for each type of port.

Port type	Speed Options (units are Mbps)
RJ-45 and RJ.5copper ports	auto (default) 10 100 1000
supported tri-speed copper SFPs	auto (default) 10 100 1000
100Mb fiber SFPs	100
1000Mb fiber SFPs	auto (default) 1000
10000Mb fiber SFP+	auto (default) 10000

Mode Interface Configuration

Default By default, ports autonegotiate speed (except for 100Base-FX ports which do not support auto-negotiation, so default to 100Mbps).

Usage Switch ports in a static or dynamic (LACP) channel group must have the same port speed and be in full duplex mode. Once switch ports have been aggregated into a channel group, you can set the speed of all the switch ports in the channel group by applying this command to the channel group.

Note Note that if multiple speeds are specified after the auto option to autonegotiate speeds, then only those speeds specified are attempted for autonegotiation.

Examples To set the speed of a tri-speed port to 100Mbps, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# speed 100
```

To return the port to auto-negotiating its speed, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# speed auto
```

To set a port to auto-negotiate its speed at 100Mbps and 1000Mbps, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# speed auto 100 1000
```

To set a port to auto-negotiate its speed at 1000Mbps only, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# speed auto 1000
```

Related Commands

duplex
ecofriendly lpi
polarity
show interface
speed (asyn)

storm-control level

Use this command to specify the threshold level for broadcasting, multicast, or destination lookup failure (DLF) traffic for the port. Storm-control limits the specified traffic type to the specified threshold.

Use the **no** variant of this command to disable storm-control for broadcast, multicast or DLF traffic.

Syntax

```
storm-control {broadcast|multicast|dlf} level <level>
no storm-control {broadcast|multicast|dlf} level
```

Parameter	Description
<level>	<0-100> Specifies the threshold as a percentage of the maximum port speed.
broadcast	Applies the storm-control to broadcast frames.
multicast	Applies the storm-control to multicast frames.
dlf	Applies the storm-control to destination lookup failure traffic.

Default By default, storm-control is disabled.

Mode Interface Configuration

Usage Flooding techniques are used to block the forwarding of unnecessary flooded traffic. A packet storm occurs when a large number of broadcast packets are received on a port. Forwarding these packets can cause the network to slow down or time out.

Example To limit broadcast traffic on `port1.0.2` to 30% of the maximum port speed, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# storm-control broadcast level 30
```

Related Commands show storm-control

switchport port-security

Enables the port-security feature. This feature is also known as the port-based learn limit. It allows the user to set the maximum number of MAC addresses that each port can learn.

Use the **no** variant of this command to disable the port-security feature.

Syntax

```
switchport port-security
no switchport port-security
```

Mode Interface Configuration

Examples To enable the port-security feature on `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# switchport port-security
```

To disable port-security feature on `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# no switchport port-security
```

switchport port-security aging

Sets the port-security MAC to time out.

Use the **no** variant of this command to set the port-security to not time out.

Syntax

```
switchport port-security aging
no switchport port-security aging
```

Mode Interface Configuration

Examples To set the MAC to time out, use the following command:

```
awplus# switchport port-security aging
```

To unset the MAC time out, use the following command:

```
awplus# no switchport port-security aging
```

switchport port-security maximum

Sets the maximum MAC address that each port can learn.

Use the **no** variant of this command to unset the maximum number of MAC addresses that each port can learn. This is same as setting the maximum number to 0. This command also resets the intrusion list table.

If a new MAC is seen on a port with port security enabled and the MAC is statically configured for another port, a violation is triggered. The maximum learn limit will be ignored and the specified intrusion action for the port will be carried out.

Syntax

```
switchport port-security maximum <0-256>
no switchport port-security maximum
```

Parameter	Description
`maximum`	Maximum number of address to learn.
<0-256>	Maximum number of address to learn.

Mode Interface Configuration

Examples To learn 3 MAC addresses on `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# switchport port-security maximum 3
```

To remove the MAC learning limit on `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# no switchport port-security maximum
```

switchport port-security violation

Sets the violation action for a switch port when the port exceeds the learning limits. The port action can be either **shutdown**, **restrict** or **protect**. If **shutdown** is set, the physical link will be disabled and "shutdown" will be shown in the config. If **restrict** is set, the packet from the un-authorized MAC will be discarded and SNMP TRAP will be generated to alert management. If **protect** is set, the packet will simply be discarded by the packet processor silently.

The **no** variant of this command sets the violation action to default. The default violation action is protect.

Syntax

```
switchport port-security violation {shutdown|restrict|protect}
no switchport port-security violation
```

Parameter	Description
`shutdown`	Disable the port.
`restrict`	Alert the network administrator.
`protect`	Discard the packet.

Mode Interface Configuration

Examples To set the action to be shutdown on `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# switchport port-security violation shutdown
```

To set the port-security action to the default (protect) on `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# no switchport port-security violation
```

thrash-limiting

Sets and configures the thrash limit action that will be applied to any port on the switch when a thrashing condition is detected. The thrash-limiting timeout specifies the time, in seconds, for which the thrash action is employed.

Syntax

```
thrash-limiting {[action {learn-disable|link-down|port-disable|
   vlan-disable|none}] [timeout <0-86400>]}

no thrash-limiting {action|timeout}
```

Parameter	Description
`action`	The mac thrashing detected action. The default is vlan-disable.
`learn-disable`	Disable mac address learning
`link-down`	Block all traffic on an interface - link down
`port-disable`	Block all traffic on an interface - link remains up
`vlan-disable`	Block all traffic on a vlan. Note that setting this parameter will also enable ingress filtering.
`none`	No thrash action
`timeout`	Set the duration for the thrash action
<0-86400>	The duration of the applied thrash action in seconds. The default is 1 seconds.

Default The default action is learn-disable.

Mode Interface Configuration

Usage See the **Thrash Limiting** section in **Chapter 16, Switching Introduction** for relevant conceptual, configuration, and overview information prior to applying this command.

Examples To set the action to learn disable for `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# thrash-limiting action learn-disable
```

To block all traffic on a vlan, use the following command:

```
awplus# configure terminal
awplus(config)# thrash-limiting action vlan-disable
```

To set the thrash limiting timeout to 5 seconds, use the following command:

```
awplus(config-if)# thrash-limiting timeout 5
```

To set the thrash limiting action to its default, use the following command:

```
awplus(config-if)# no thrash-limiting action
```

To set the thrash limiting timeout to its default, use the following command:

```
awplus(config-if)# no thrash-limiting timeout
```

Related Commands

loop-protection
loop-protection action
loop-protection timeout
show loop-protection

undebug loopprot

This command applies the functionality of the **no debug loopprot** command on page 17.8.

undebug platform packet

This command applies the functionality of the **no debug platform packet** command on page 17.9.

Chapter 18: VLAN Introduction

VLANs Introduction 18.2
Virtual LANs (VLANs) 18.2
Configuring VLANs 18.3
VLAN Double Tagging (VLAN Stacking) 18.5
How Double-Tagged VLANs Work 18.5
VLAN Rules for Double Tagging 18.5
Restrictions when Using Double-Tagged VLANs 18.6
Configuring Double-Tagged VLANs 18.6
Private VLANs 18.11
Private VLANs for Ports in Access Mode 18.11
Private VLAN Operation with Ports in Access Mode 18.13
Access Mode Private VLAN Configuration Example 18.14
Private VLANs for Trunked Ports 18.17
Trunked Port Private VLAN Configuration Example 18.18
Protocol Based VLAN Configuration Example 18.22
VLAN Statistics 18.25
Counter Operation 18.25

VLANs Introduction

This chapter describes Virtual LANs (VLAN), VLAN features and configuration on the switch. For detailed descriptions of commands used to configure VLANs, see **Chapter 19, VLAN Commands**. For information about Voice VLAN and LLDP-MED, see **Chapter 96, LLDP, LLDP-MED and Voice VLAN Introduction and Configuration**.

Virtual LANs (VLANs)

A Virtual LAN (VLAN) is a logical, software-defined subnetwork. It allows similar devices on the network to be grouped together into one broadcast domain, irrespective of their physical position in the network. Multiple VLANs can be used to group workstations, servers, and other network equipment connected to the switch, according to similar data and security requirements.

Decoupling logical broadcast domains from the physical wiring topology offers several advantages, including the ability to:

- Move devices and people with minimal, or no, reconfiguration
- Change a device's broadcast domain and access to resources without physically moving the device, by software reconfiguration or by moving its cable from one switch port to another
- Isolate parts of the network from other parts, by placing them in different VLANs
- Share servers and other network resources without losing data isolation or security
- Direct broadcast traffic to only those devices which need to receive it, to reduce traffic across the network
- Connect 802.1Q-compatible switches together through one port on each switch

Devices that are members of the same VLAN only exchange data with each other through the switch's Layer 2 switching capabilities. To exchange data between devices that are located in different VLANs, the switch's Layer 3 (routing) capabilities are used.

Different IP subnets are associated with different VLANs. The switch's IP router table will be populated by the routes to the subnets on any active VLANs, and by routes statically configured over active VLAN interfaces, or learned via routing protocols operating over these interfaces.

The device supports up to 4094 VLANs (the maximum allowed by the VID field in the 802.1Q tag). On some devices a few of these VLANs may be reserved for management purposes.

When the switch is first powered up (and therefore unconfigured), it creates a default VLAN with a VID of 1 and an interface name of *vlan1*. In this initial condition, the switch attaches all its ports to this default VLAN.

The default VLAN cannot be deleted, and ports can only be removed from it if they also belong to at least one other VLAN. If all the devices on the physical LAN belong to the same logical LAN, that is, the same broadcast domain, then the default settings will be acceptable, and no additional VLAN configuration is required.

Configuring VLANs

Defaults

By default, all switch ports are in access mode, are associated with the default VLAN (**vlan1**), and have ingress filtering on. You cannot delete **vlan1**.

VLAN names

When you create a VLAN (using the **vlan** command), you give it a numerical VLAN Identifier (VID) - a number from 2 to 4094. If tagged frames are transmitted from this VLAN, they will contain this VID in their tag. You may also give it an arbitrary alphanumeric name containing a meaningful description, which is not transmitted to other devices.

When referring to a VLAN, some commands require the VLAN to be specified by its VID while some commands require it to be specified by its interface name: vlan<VID>. In command output, the VLAN may be referred to by its VID, its interface name (vlan<VID>), or its VLAN name (the arbitrary alphanumeric string).

You can name a VLAN with a string containing “vlan” and its VLAN Identifier (VID). To avoid confusion, we recommend not naming it “vlan” followed by any number different from its VID.

Access mode

A switch port in access mode sends untagged Ethernet frames, that is, frames without a VLAN tag. Each port is associated with one VLAN (the port-based VLAN, by default, *vlan1*), and when it receives untagged frames, it associates them with the VID of this VLAN. You can associate the port with another VLAN (using the **switchport access vlan** command). This removes it from the default VLAN.

Use access mode for any ports connected to devices that do not use VLAN tagging, for instance PC workstations.

Trunk mode

A switch port in trunk mode is associated with one or more VLANs for which it transmits VLAN-tagged frames, and for which it identifies incoming tagged frames with these VIDs.

To allow a switch port to distinguish and identify traffic from different VLANs, put it in trunk mode (using the **switchport mode trunk** command), and add the VLANs (using the **switchport trunk allowed vlan** command). Use trunk mode for ports connected to other switches which send VLAN-tagged traffic from one or more VLANs.

A trunk mode port may also have a native VLAN (by default vlan1), for which it transmits untagged frames, and with which it associates incoming untagged frames (using the **switchport trunk native vlan** command).

Ports in trunk mode can be enabled as promiscuous ports for private VLANs (using the **switchport mode private-vlan trunk promiscuous**) and secondary ports for private VLANs (using the **switchport mode private-vlan trunk secondary**).

Mirror ports

A mirror port cannot be associated with a VLAN. If a switch port is configured to be a mirror port (using the **mirror interface** command), it is automatically removed from any VLAN it was associated with.

VLANs and channel groups

All the ports in a channel group must have the same VLAN configuration: they must belong to the same VLANs and have the same tagging status, and can only be operated on as a group.

Table 18-1: Configuration procedure for VLANs

Step 1. Create VLANs	
`awplus#` **`configure terminal`**	Enter Configuration mode.
`awplus(config)#` `vlan database`	Enter VLAN Configuration mode.

`awplus(config-vlan)#`
`vlan <vid> [name <vlan-name>] [state {enable|disable}]`
or
`vlan <vid-range> [state {enable|disable}]`
	Create VLANs.

Step 2. Associate switch ports with VLANs	
`awplus(config-vlan)#` `interface <port-list>`	Associate switch ports in access mode with VLANs:
`awplus(config-if)#` `switchport access vlan <vlan-id>`	Enter Interface Configuration mode for the switch ports that will be in access mode for a particular VLAN. Associate the VLAN with these ports in access mode. Repeat for other VLANs and ports in access mode.
`awplus(config-if)#` `interface <port-list>`	Associate switch ports in trunk mode with VLANs. Enter Interface Configuration mode for all the switch ports that will be in trunk mode for a particular set of VLANs.
`awplus(config-if)#` `switchport mode trunk [ingress-filter {enable|disable}]`	Set these switch ports to trunk mode.

`awplus(config-if)#`
`switchport trunk allowed vlan all`
or
`switchport trunk allowed vlan add <vid-list>`
	Allow these switch ports to trunk this set of VLANs.

`awplus(config-if)#` `switchport trunk native vlan {<vid>|none}`	By default, a trunk mode switch port's native VLAN, the VLAN that the port uses for untagged packet, is VLAN 1. If required, change the native VLAN from the default.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `exit`	Return to Privileged Exec mode.
`awplus#` `show vlan {all|brief|dynamic|static|auto|static-ports<1-4094>}`	Confirm VLAN configuration.

VLAN Double Tagging (VLAN Stacking)

VLAN double tagging, also known as VLAN Stacking, Nested VLANs, or Q-in-Q VLANs, is used to operate a number of private Layer 2 networks within a single public Layer 2 network. This feature provides simple access infrastructure for network service providers to operate Metropolitan Area Networks (MANs) as commercial value added networks. Customers can connect to a service provider's network at multiple locations and use their own VLAN IDs, without requiring the service provider's equipment between to know about those VLANs.

A nested VLAN implementation consists of the following port types:

- Provider ports - these connect to a service provider's Layer 2 network
- Customer edge ports - these connect to a customer's private Layer 2 network

How Double-Tagged VLANs Work

In a nested VLAN environment VLAN tagging exists at two levels:

- client tagging (C-tag)
- service provider tagging (S-tag)

When nested VLAN functionality is enabled, the service provider assigns to each of its clients an individual 12 bit customer VID called an S-Tag. The S-Tag field has an identical structure to a conventional VLAN tag field.

The S-Tag is attached to a packet as it enters the service provider network at the customer edge port. From this point on, the S-Tag is used for transmission within the service provider, or public Layer 2, network. The S-Tag is then removed as it leaves the destination customer edge port. This process is shown in **Figure 18-1 on page 18.7**.

The VID that is used within the client's own network, the C-Tag, is ignored by the service provider network and bridging is based on the value of the S-Tag. The ethertype of the S-Tag is set by changing the Tag Protocol Identifier (TPID). Once the S-Tag is removed from the packet, it is forwarded "as is" out of the customer-edge port. The tagged status of the Customer port is ignored on egress.

VLAN Rules for Double Tagging

When double-tagged VLANs are created on the switch:

- a nested VLAN belongs to only one customer and can have multiple customer-edge ports
- a port must be either a customer-edge port or a provider port, but cannot be both

A service provider port:

- accepts only tagged packets
- transmits only tagged packets
- can be in many double-tagged VLANs

A customer edge port:

- accepts both tagged and untagged packets
- transmits both tagged and untagged packets
- can be a member of only one nested VLAN

Restrictions when Using Double-Tagged VLANs

Restrictions when double-tagged VLANs are implemented are:

- Ethernet bridging is based on the S-Tag VID instead of the packet C-Tag VID. The packets C-Tag VID does not change
- ARP packet trapping is restricted
- hardware filtering does not work above MAC address level

Configuring Double-Tagged VLANs

You need a special feature license to use double-tagged VLANs. Contact your authorized Allied Telesis distributor or reseller for more information.

Set the Tag Protocol Identifier (TPID)

If required, you can change the Tag Protocol Identifier (TPID) from its default (for VLAN stacking) of 0x8100 (specified as hex notation), with the **platform vlan-stacking-tpid** command on page 17.28. Note that this command specifies the TPID value that applies to all VLANs used for double-tagged VLANs. You cannot set individual TPID values for different VLANs within a multi double-tagged VLAN network

Set the Maximum Receive Unit (MRU)

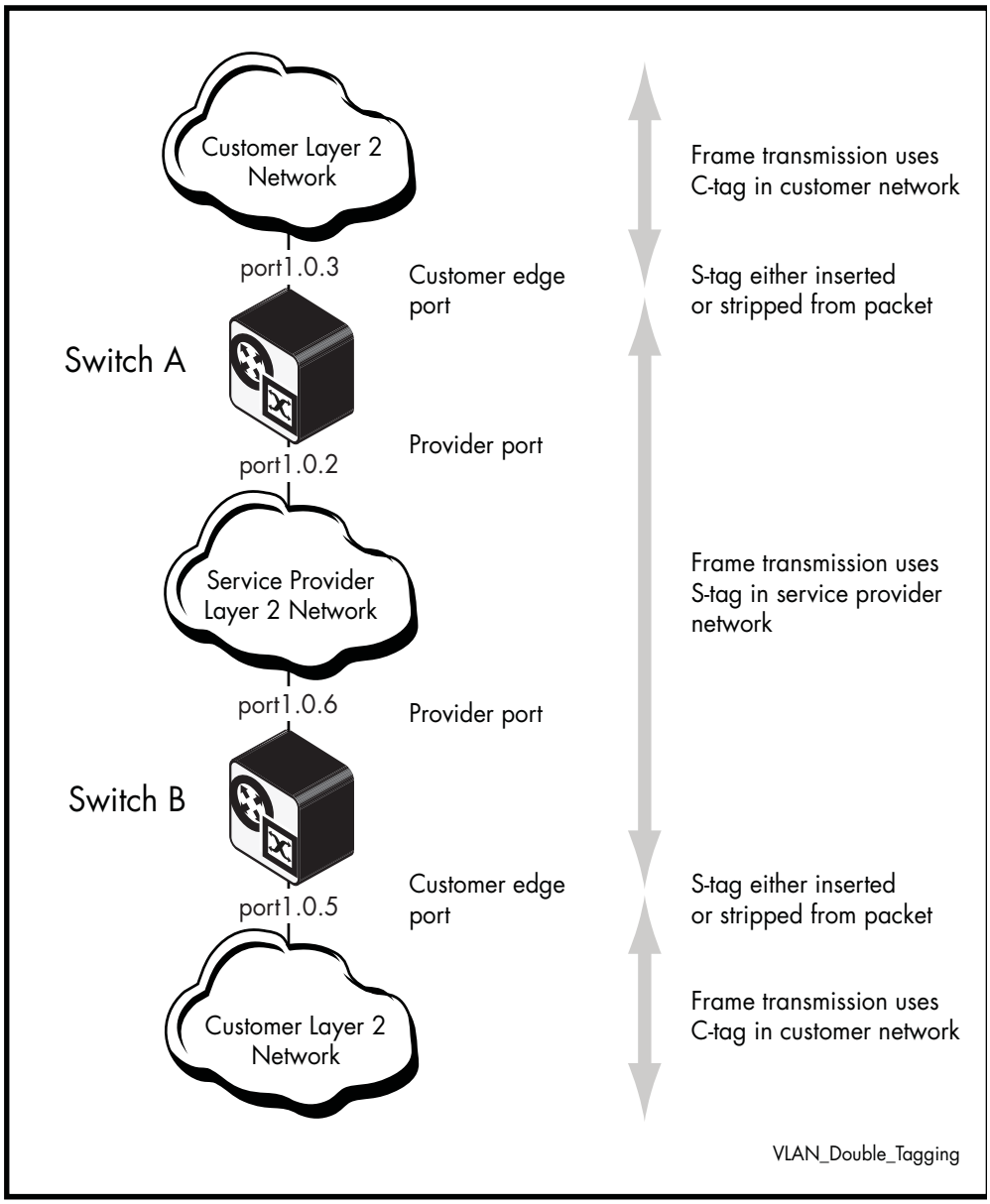
Adding the S-Tag can result in frame sizes that exceed the maximum of 1522 bytes. In order to cope with these larger than normal frames, you should increase the MRU size set for ports configured for double-tagged VLANs. Set the MRU size to:

- 9710 bytes for ports that work at speeds of either 100 Mbps or 100 Mbps
- 10240 bytes for ports that work at speeds of 1000 Mbps

For more information, see the **mru** command on page 14.5.

Double-tagged VLAN configuration example

Figure 18-1: VLAN double tagging



Step 1. Create and enable the service provider VLAN 2 (the VLAN that will be used in the outer-tag)	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `vlan database`	Enter VLAN database mode.
`awplus(config-vlan)#` `vlan 2 state enable`	Create and enable VLAN 2.
`awplus(config-vlan)#` `exit`	Return to Global Configuration mode.

Step 2. Configure port 1.0.2 as a provider-port member of VLAN 2	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `interface port1.0.2`	Select `port1.0.2` for configuring.
`awplus(config-if)#` `switchport mode trunk`	Set the port to trunk mode.
`awplus(config-if)#` `switchport trunk allowed vlan add 2`	Add the VLAN to be trunked over the port.
`awplus(config-if)#` `switchport vlan-stacking provider-port`	Enable VLAN stacking and set the port to be a provider port.
Step 3. Set the Maximum Receive Unit (MRU)	
`awplus(config-if)#` `mru <10240>`	Specify the MRU size in bytes.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
Step 4. Configure port 1.0.3 as a customer edge port member of VLAN 10	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `interface port1.0.3`	Select `port1.0.3` for configuring.
`awplus(config-if)#` `switchport mode access`	Set the port to access mode.
`awplus(config-if)#` `switchport access vlan 2`	Associate the port with VLAN 2.
`awplus(config-if)#` `switchport vlan-stacking customer-edge-port`	Enable VLAN stacking and set the port to be a customer edge port.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.

Table 18-2: Switch B Configuration

Step 5. Create and enable the service provider VLAN 2 (the VLAN that will be used in the outer-tag)	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter VLAN database mode.
`awplus(config-vlan)#`	
`vlan 2 state enable`	Create and enable VLAN 2.
`awplus(config-vlan)#`	
`exit`	Return to Global Configuration mode.
Step 6. Configure port 1.0.6 as a provider-port member of VLAN 2	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`interface port1.0.6`	Select `port1.0.6` for configuring.
`awplus(config-if)#`	
`switchport mode trunk`	Set the port to trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 2`	Add the VLAN to be trunked over the port.
`awplus(config-if)#`	
`switchport vlan-stacking provider-port`	Enable VLAN stacking and set the port to be provider port.
Step 7. Set the Maximum Receive Unit (MRU)	
`awplus(config-if)#`	
`mru <10240>`	Specify the MRU size in bytes.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
Step 8. Configure port 1.0.5 as a customer edge port member of VLAN 10	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`interface port1.0.5`	Select `port1.0.5` for configuring.
`awplus(config-if)#`	
`switchport mode access`	Set the port to access mode.

Table 18-2: Switch B Configuration(cont.)

`awplus(config-if)#`	
`switchport access vlan 2`	Associate the port with VLAN 2.
`awplus(config-if)#`	
`switchport vlan-stacking customer-edge-port`	Enable VLAN stacking and set the port to be a customer edge port.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.

Private VLANs

Private VLANs combine the network advantages of conventional VLANs, with an added degree of privacy obtained by limiting the connectivity between selected ports.

This section provides an introduction to:

- Private VLANs for ports in access mode
- **Private VLANs for Trunked Ports**

Private VLANs for Ports in Access Mode

An example application of a private VLAN would be a library in which user booths each have a PC with Internet access. In this situation it would usually be undesirable to allow communication between these individual PCs. Connecting the PC to ports within a private isolated VLAN would enable each PC to access the Internet or a library server via a single connection, whilst preventing access between the PCs in the booths.

Another application might be to use private VLANs to simplify IP address assignment. Ports can be isolated from each other whilst still belonging to the same subnet.

A private VLAN comprises the following components:

- **a single promiscuous port**
- **one or more host ports.** There are two types of host port:
  - « **isolated ports**: These can only communicate with the promiscuous port that is associated with the isolated VLAN.
  - « **community ports:** These can communicate with their associated promiscuous port and other community ports within the community VLAN.
- **a single primary VLAN**
- **one or more secondary VLANs:** There are two types of secondary VLAN:
  - « **isolated VLANs:** In this VLAN type, communication can only take place between each host port and its associated promiscuous port.
  - « **community VLANs:** In this VLAN type, communication can take place between host ports and between each host port and its associated promiscuous port.

Membership rules for private VLANs in access mode

The following membership rules apply when creating and operating private VLANs in access mode.

Each private VLAN:

- must contain one promiscuous port (or aggregated link)
- may contain multiple host ports
- can be configured to span switch instances
- can only contain promiscuous and host ports
- cannot use the default VLAN (vlan1)
- a private *isolated* VLAN can only contain a single promiscuous port
- a private *community* VLAN can contain more than one promiscuous port

A promiscuous port:

- is a member of the primary VLAN and all its associated secondary VLANs
- cannot be a member of both private and non-private VLANs

A host port:

- can be a member of multiple private (community) VLANs, but all these VLANs must share the same promiscuous port
- cannot be a host port in some VLANs and a non-host port in others
- cannot be a promiscuous port in another VLAN

Promiscuous ports

A promiscuous port can communicate with all ports that are members of its associated secondary VLANs. Multiple promiscuous ports can exist in a primary VLAN, but only if the primary VLAN is only associated with community VLANS (that is, that there are no isolated VLANs associated with this port).

A promiscuous port is a member of the primary VLAN and all associated secondary VLANs. Its Port VID is set to the VLAN ID of the primary VLAN.

Host ports

Host ports have two levels of connectivity depending on whether they exist in an isolated or a community VLAN.

1. Host ports within an isolated VLAN

These ports are only allowed to communicate with their VLAN's promiscuous port, even though they share their secondary (isolated) VLAN with other hosts. The host ports receive their data from the promiscuous port via the primary VLAN, and individually transmit their data to the promiscuous port via their common secondary VLAN.

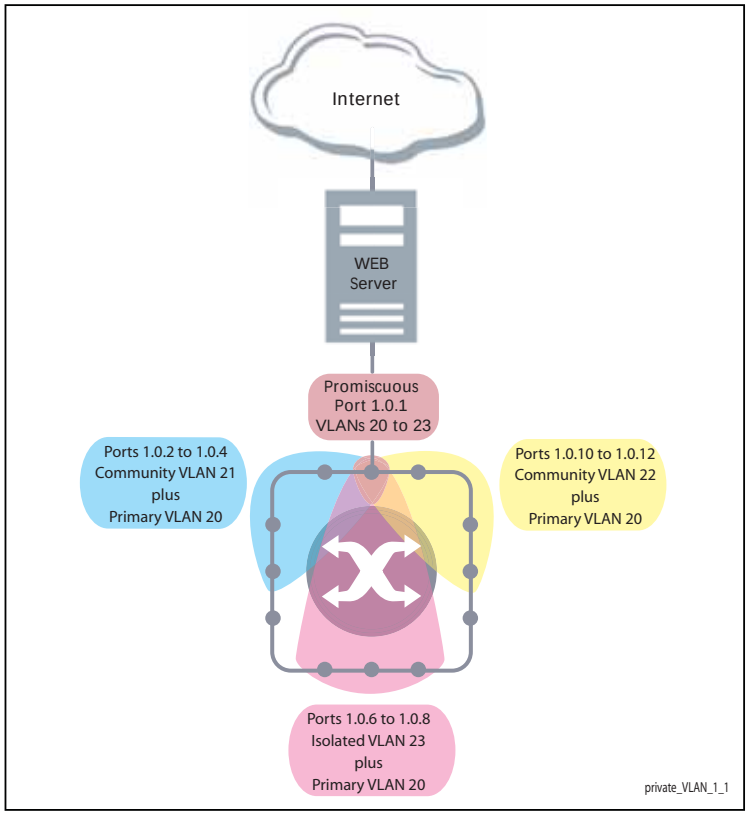
2. Host ports within a community VLAN

These ports are able to communicate with both the promiscuous port and the other ports within the community VLAN that they are associated with. They receive their data from the promiscuous port via the primary VLAN, and transmit their data to both the promiscuous port and the other host ports (within their community VLAN) via their common secondary VLAN. However, the only external path from a community VLAN is from its promiscuous port.

Private VLAN Operation with Ports in Access Mode

A basic private VLAN operation is shown in the following figure. It comprises a primary VLAN 20 plus community and isolated VLANS.

Figure 18-2: Private VLAN



The ports on this switch have the following configuration:

- Port 1.0.1 is the promiscuous port and is a member of the primary VLAN 20 and all its associated secondary VLANs.
- Ports 1.0.2 to 1.0.4 are members of the community VLAN 21 and are able to communicate with both the promiscuous port and all other ports in VLAN 21.
- Ports 1.0.10 to 1.0.12 are members of the community VLAN 22 and are able to communicate with both the promiscuous port and all other ports in VLAN 22.
- Ports 1.0.6 to 1.0.8 are members of the isolated VLAN 23. Each of these ports can only communicate with the promiscuous port.

Table 18-3: Private VLANs - Port Tagging

Port	Mode	Untagged VLAN Membership	PVID
1.0.1	Promiscuous	20, 21, 22, 23	20
1.0.2 to 1.0.4	Host	20, 21	21
1.0.10 to 1.0.12	Host	20, 22	22
1.0.6 to 1.0.8	Host	20, 23	23
1.0.5	Not members of the private VLAN		-
1.0.9	Not members of the private VLAN		-

Private VLANs operate within a single switch and comprise one primary VLAN plus a number of secondary VLANS. All data enters the private VLAN ports untagged.

Using the example of **Figure 18-2**, data enters the switch via the promiscuous `port1.0.1` and is forwarded to the host ports using VLAN 20, the primary VLAN. Data returning from the host ports to the promiscuous port (and exiting the switch) use the secondary VLAN associated with its particular host port, VLAN 21, 22, or 23 in the example.

Thus the data flows into the switch via the primary VLAN and out of the switch via the secondary VLANs. This situation is not detected outside of the switch, because all its private ports are untagged. Note however, that data flowing between ports within the same community VLAN will do so using the VID of the community VLAN.

Portfast on private VLANS

Within private VLANs, we recommend that you place all host ports into spanning-tree portfast mode and enable BPDU guard. Portfast assumes that because host ports will also be edge ports, they will have no alternative paths (loops) via other bridges. These ports are therefore allowed to move directly from the spanning-tree blocking state into the forwarding state, thus bypassing the intermediate states.

Applying BPDU guard is an extra precaution. This feature disables an edge port if it receives a BPDU frame, because receiving such a frame would indicate that the port has a connection to another network bridge.

For more information on BPDU guard and portfast, see their following commands:

- **spanning-tree portfast bpdu-guard** command on page 21.60
- **spanning-tree portfast (STP)** command on page 21.56

Access Mode Private VLAN Configuration Example

Table 18-4: Configuration procedure for access mode private VLANs

Command	Description
Step 1. Create the VLANs	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `vlan database`	Enter VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 20-23`	Create the VLANs.
Step 2. Create the private VLANs and set the type	
`awplus(config-vlan)#` `private-vlan 20 primary`	Create primary VLAN 20.
`awplus(config-vlan)#` `private-vlan 21 community`	Create community VLAN 21.
`awplus(config-vlan)#` `private-vlan 22 community`	Create community VLAN 22.

Table 18-4: Configuration procedure for access mode private VLANs (cont.)

Command	Description
`awplus(config-vlan)#` `private-vlan 23 isolated`	Create isolated VLAN 23.
Step 3. Associate the secondary VLANs with the primary VLAN	
`awplus(config-vlan)#` `private-vlan 20 association add 21`	Associate secondary VLAN 21 with the primary VLAN 20.
`awplus(config-vlan)#` `private-vlan 20 association add 22`	Associate secondary VLAN 22 with the primary VLAN 20.
`awplus(config-vlan)#` `private-vlan 20 association add 23`	Associate secondary VLAN 23 with the primary VLAN 20.
Step 4. Set port 1.0.1 to be the promiscuous port	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.1`	Enter Interface Configuration mode for `port1.0.1`.
`awplus(config-if)#` `switchport mode private-vlan promiscuous`	Set the port as a promiscuous ports.
Step 5. Set the other ports to be host ports	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.2-1.0.4, port1.0.6`	Enter Interface Configuration mode for the ports.
`awplus(config-if)#` `switchport mode private-vlan host`	Set the ports as host ports.
Step 6. On the promiscuous port, map the primary VLAN to each of the secondary VLANs	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.1`	Enter Interface Configuration mode for `port1.0.1`.

Table 18-4: Configuration procedure for access mode private VLANs (cont.)

Command	Description
`awplus(config-if)#` `switchport private-vlan mapping 20 add 21-23`	Associate primary VLAN 20 and the secondary VLANs 21 to 23 to the promiscuous port.
Step 7. Associate the community host ports with the community VLANs	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.2-1.0.4`	Enter Interface Configuration mode for ports 1.0.2 to 1.0.4.
`awplus(config-if)#` `switchport private-vlan host-association 20 add 21`	Associate primary VLAN 20 and secondary VLAN 21 to the host ports.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.10-1.0.12`	Enter Interface Configuration mode for ports 1.0.10 to 1.0.12.
`awplus(config-if)#` `switchport private-vlan host-association 20 add 22`	Associate primary VLAN 20 and secondary VLAN 22 to the host ports.
Step 8. Associate the isolated host ports with the isolated VLAN 23	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.6-1.0.8`	Enter Interface Configuration mode for ports 1.0.6 to 1.0.8.
`awplus(config-if)#` `switchport private-vlan host-association 20 add 23`	Associate primary VLAN 20 and secondary VLAN 23 to the host ports.

Private VLANs for Trunked Ports

Private VLAN trunk ports allow you to combine traffic for private isolated VLANs over a trunk. A port in trunk mode enabled as a promiscuous port with the **switchport mode private-vlan trunk promiscuous** command can carry both multiple isolated private VLANs and non-private VLANs. A promiscuous port in trunk mode allows you to combine multiple isolated VLANs on a single trunk port. A port in trunk mode enabled as a secondary port with the **switchport mode private-vlan trunk secondary** command can combine traffic for multiple isolated VLANs over a trunk.

Note Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. Private VLAN trunk ports and GVRP are mutually exclusive.

A private VLAN group for trunked ports comprises the following components:

- **a single promiscuous port**
- **one or more isolated secondary ports:** These can only communicate with the associated promiscuous port.
- **isolated VLANs:** In this VLAN type, communication can only take place between each secondary port and its associated promiscuous port. Membership rules for private VLANs for trunked ports

The following membership rules apply when creating and operating private VLANs for trunked ports.

A promiscuous trunk port:

- must be in trunk mode
- can be a member of both isolated VLANs and non-isolated VLANs
- has a group ID that is solely used to associate the promiscuous port with secondary ports

A secondary trunk port:

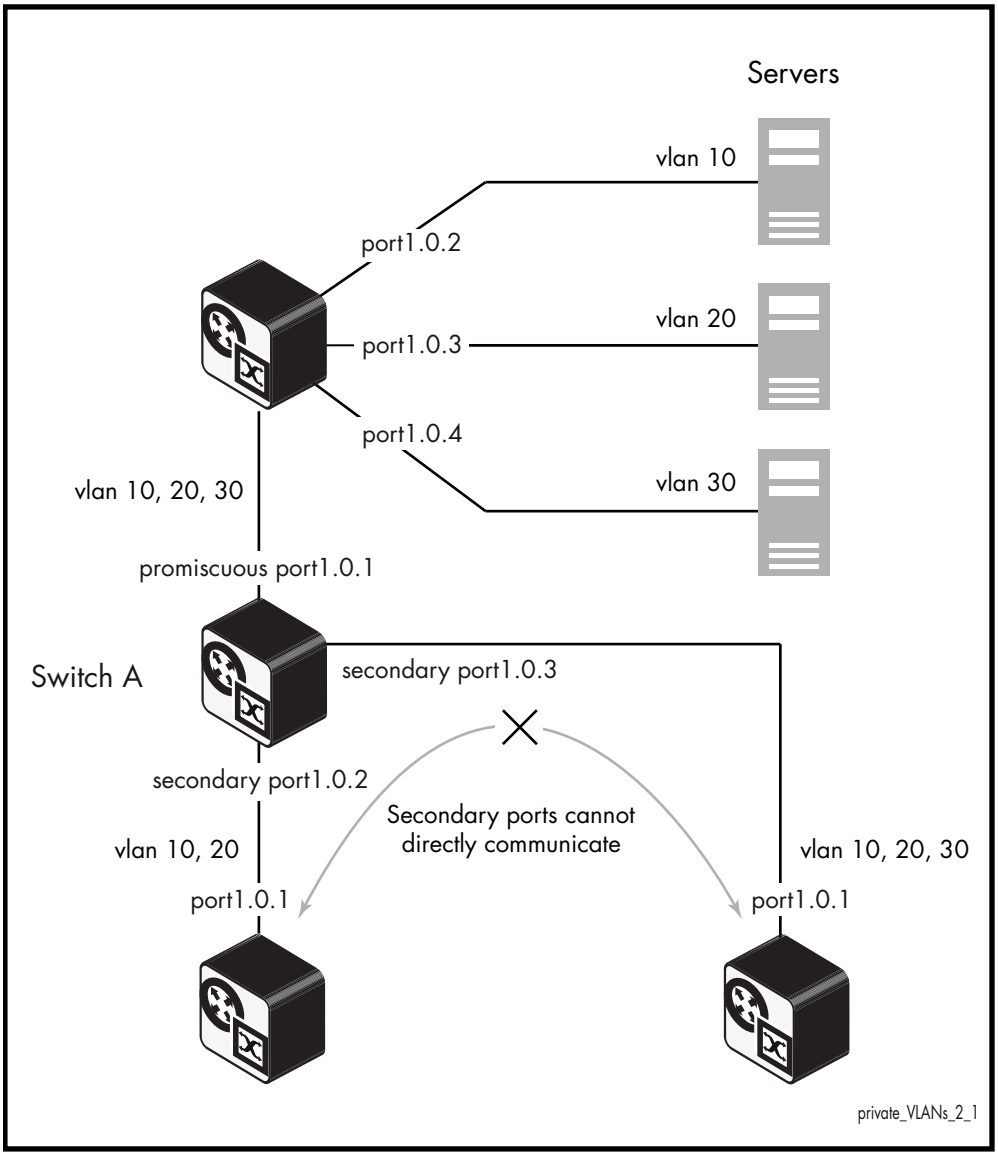
- must be in trunk mode
- can only be a member of isolated VLANs
- cannot be a promiscuous port in another VLAN
- has a group ID that is solely used to associate the secondary port with its promiscuous port

Unlike private VLANs for ports in access mode, private VLANs for trunked ports have no secondary to primary VLAN mappings.

Trunked Port Private VLAN Configuration Example

A basic trunked port private VLAN operation is shown in **Figure 18-3**.

Figure 18-3: Trunked port private VLAN



The ports on **Switch A** have the following configuration:

- Port 1.0.1 is the promiscuous port, and has a group ID of 1
- Port 1.0.2 is a secondary port for isolated private VLANs 10 and 20, and has a group ID of 1
- Port 1.0.3 is a secondary port for isolated private VLANs 10, 20 and 30, and has a group ID of 1

The configuration procedure in **Table 18-5** show the steps to configure **Switch A**.

Table 18-5: Configuration procedure for Switch A

Command	Description
Step 1. Create the VLANs	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `vlan database`	Enter VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 10,20,30`	Create the VLANs.
Step 2. Create the private VLANs and set the type	
`awplus(config-vlan)#` `private-vlan 10 isolated`	Create isolated VLAN 10.
`awplus(config-vlan)#` `private-vlan 20 isolated`	Create isolated VLAN 20.
`awplus(config-vlan)#` `private-vlan 30 isolated`	Create isolated VLAN 30.
Step 3. Set port 1.0.1 to trunk mode and add the VLANs to be trunked over the port	
`awplus(config-vlan)#` `interface port1.0.1`	Enter Interface Configuration mode for `port1.0.1`.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of the port to trunk.
`awplus(config-if)#` `switchport trunk allowed vlan add 10,20,30`	Add the VLANs to be trunked over this port.
Step 4. Set port 1.0.2 to trunk mode and add the VLANs to be trunked over the port	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.2`	Enter Interface Configuration mode for `port1.0.2`.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of the port to trunk.
`awplus(config-if)#` `switchport trunk allowed vlan add 10,20`	Add the VLANs to be trunked over this port.

Table 18-5: Configuration procedure for Switch A(cont.)

Command	Description
Step 5. Set port 1.0.3 to trunk mode and add the VLANs to be trunked over the port	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.3`	Enter Interface Configuration mode for port 1.0.3.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of the port to trunk.
`awplus(config-if)#` `switchport trunk allowed vlan add 10,20,30`	Add the VLANs to be trunked over this port.
Step 6. Set port 1.0.1 to be the promiscuous port	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.1`	Enter Interface Configuration mode for port 1.0.1.
`awplus(config-if)#` `switchport mode private-vlan trunk promiscuous group 1`	Enable the port in trunk mode to be promiscuous port for isolated VLANs 10, 20 and 30 with a group ID of 1.
Step 7. Set port 1.0.2 to be a secondary port	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface port1.0.2`	Enter Interface Configuration mode for `port1.0.2`.
`awplus(config-if)#` `switchport mode private-vlan trunk secondary group 1`	Enable the port in trunk mode to be a secondary port for isolated VLANs 10 and 20 with a group ID of 1.
Step 8. Set port 1.0.3 to be a secondary port	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.

Table 18-5: Configuration procedure for Switch A(cont.)

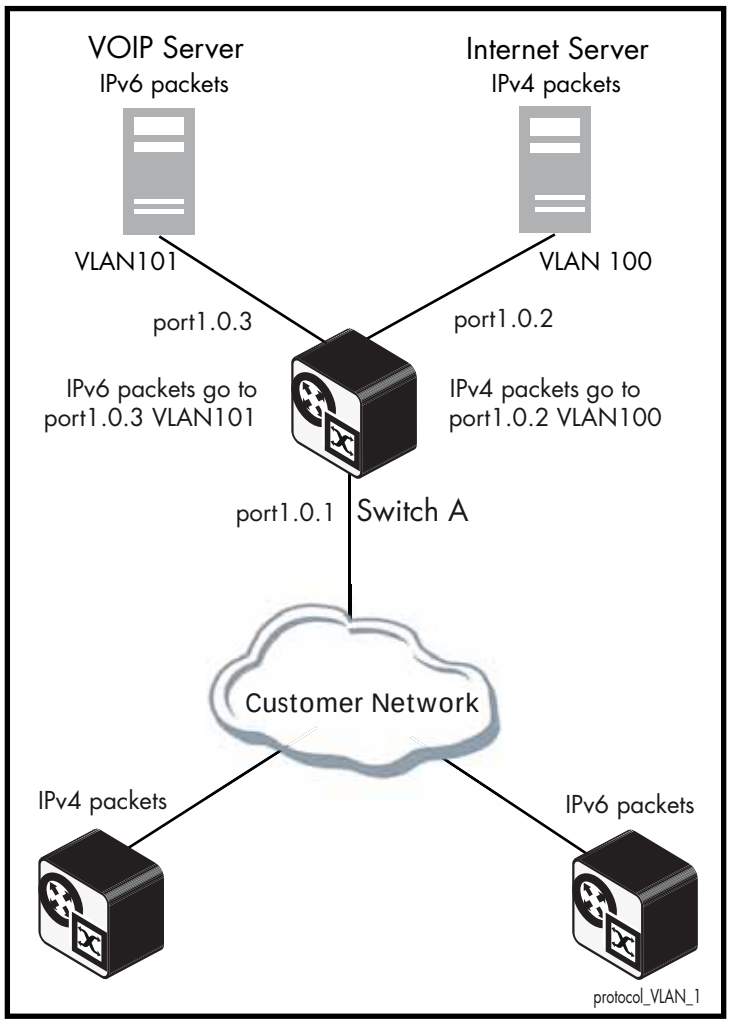
Command	Description
`awplus(config)#`	
`interface port1.0.3`	Enter Interface Configuration mode for `port1.0.3`.
`awplus(config-if)#`	
`switchport mode private-vlan trunk secondary group 1`	Enable the port in trunk mode to be a secondary port for isolated VLANs 10, 20 and 30 with a group ID of 1.

Protocol Based VLAN Configuration Example

A protocol based VLAN topology is shown below in **Figure 18-4**.

See the configuration procedure to configure **Switch A** in **Table 18-5** on the next page.

Figure 18-4: Protocol based VLAN configuration



Switch A has the following configuration to enable protocol based VLAN classification:

- VLAN 100 and VLAN 101 created and applied to port1.0.2 and port1.0.3 respectively
- IPv4 and IPv6 VLAN classifier rules created and mapped to VLAN 100 and VLAN 101
- VLAN classifier group created and mapped to port1.0.1
- VLAN 100 and VLAN 101 are trunked over port1.0.2 and port1.0.3 respectively
- IPv4 packets received on port1.0.1 go to port1.0.2 VLAN 100
- IPv6 packets received on port1.0.1 go to port1.0.3 VLAN 101

The configuration procedure in **Table 18-5** show the steps to configure **Switch A**.

Table 18-6: Configuration procedure for Switch A

Command	Description
Step 1. Create the VLANs 100 and 101	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `vlan database`	Enter VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 100,101`	Create the VLANs.
`awplus(config-vlan)#` `exit`	Exit VLAN Configuration mode.
Step 2. Create two protocol type based VLAN classifier rules for IPv4 and IPv6 mapped to VLAN 100 and 101	
`awplus(config)#` `vlan classifier rule 1 proto ip encap ethv2 vlan 100`	Create a VLAN classifier rule `1` for IPv4 packets on VLAN `100`.
`awplus(config)#` `vlan classifier rule 2 proto ipv6 encap ethv2 vlan 101`	Create a VLAN classifier rule `2` for IPv6 packets on VLAN `101`.
Step 3. Create a group of VLAN classifier rules and map the defined VLAN classifier rules 1 and 2 to the group	
`awplus(config)#` `vlan classifier group 1 add rule 1`	Add VLAN classifier rule `1` to VLAN classifier group `1`.
`awplus(config)#` `vlan classifier group 1 add rule 2`	Add VLAN classifier rule `2` to VLAN classifier group `1`.
Step 4. Associate the created VLAN classifier group 1 with port1.0.1	
`awplus(config)#` `interface port1.0.1`	Enter Interface Configuration mode for `port1.0.1`.
`awplus(config-if)#` `vlan classifier activate 1`	Associate VLAN classifier group `1` with `port1.0.1`.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode.
Step 5. Add VLAN 100 to be trunked over port1.0.2	

Table 18-6: Configuration procedure for Switch A(cont.)

Command	Description
`awplus(config)#` `interface port1.0.2`	Enter Interface Configuration mode for `port1.0.2`.
`awplus(config-if)#` `switchport mode trunk`	Enable switchport trunking on `port1.0.2`.
`awplus(config-if)#` `switchport trunk allowed vlan add 100`	Add VLAN 100 to be trunked over `port1.0.2`.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode.
Step 6. Add VLAN 101 to be trunked over port1.0.3	
`awplus(config)#` `interface port1.0.3`	Enter Interface Configuration mode for `port1.0.3`.
`awplus(config-if)#` `switchport mode trunk`	Enable switchport trunking on `port1.0.3`.
`awplus(config-if)#` `switchport trunk allowed vlan add 101`	Add VLAN 101 to be trunked over `port1.0.3`.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode.

VLAN Statistics

This feature provides a series of data counters each able to count both the number of received frames or the number of received bytes (octets) belonging to a particular VLAN. Data frames are counted as they enter the switch ports. By allocating VLANs to each customer, a service provider could use the VLAN counter output to provide the basis for a traffic based billing component.

Counter Operation

Two scenarios are detailed: in the first scenario the switch is being used at the edge of the network; in the second scenario it is directly connected to an edge switch. In each situation, separate counters are maintained for incoming traffic that is associated with a particular VLAN across a range of ports. This enables both incoming and outgoing traffic volumes to be measured.

A port may not be assigned to multiple counter instances so as to count frames (or bytes) within the same VLAN.

The byte count includes frame headers, therefore the byte counter for a VLAN tagged frame will be 4 bytes longer than for an untagged frame.

Where a VLAN packet counter instance encompasses ports on a stacked member and the member is removed from the stack, these ports will automatically be removed from the counter instance. If this process removes all ports within a counter instance, then the instance will be deleted.

Edge switch scenario

This network is shown in **Figure 18-5 on page 18.26**. The total data count is the upload count plus the download packet count

Customer A data count

The upload data count for customer A is determined by monitoring the inbound VLAN 10 packets on ports 1.0.2 and 1.0.8 (i.e. packets from Customer A's network). These ports must be untagged members of VLAN 10. Note that packets traveling between these ports will be included in the count.

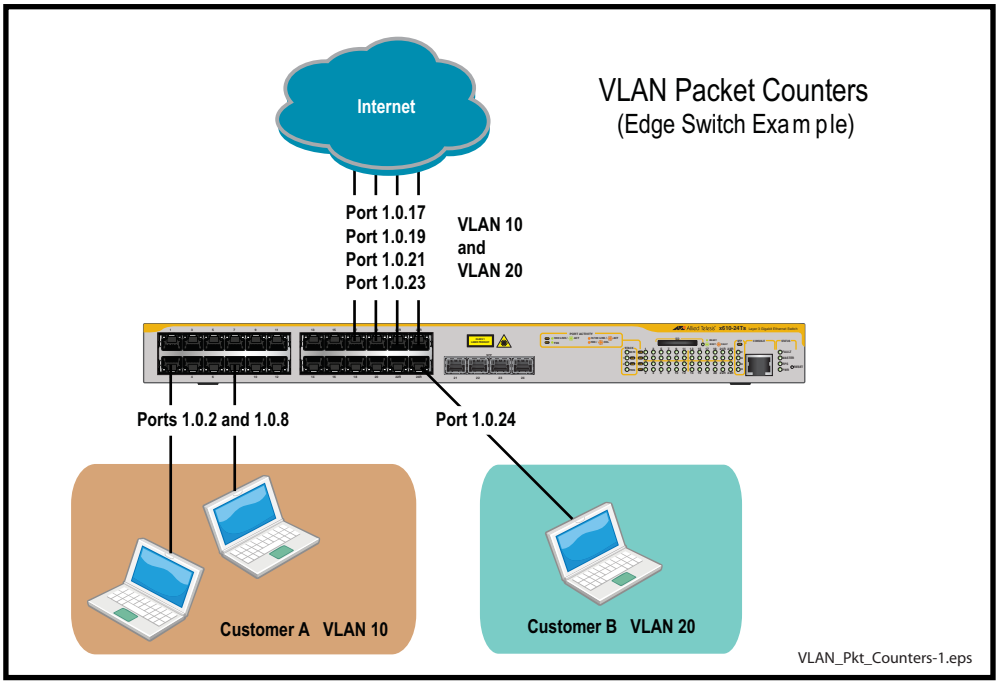
The download data count for customer A is determined by monitoring inbound VLAN 10 packets on ports 1.0.17, 1.0.19, 1.0.21, and 1.0.23 (i.e. packets destined for Customer A's network from the Internet).

Customer B data count

The upload data count for customer B is determined by monitoring the inbound VLAN 20 packets on `port1.0.24` (i.e. packets from Customer B's network). This port must be an untagged member of VLAN 20.

The download data count for customer B is determined by monitoring inbound VLAN 20 packets on ports 1.0.17, 1.0.19, 1.0.21, and 1.0.23 (i.e. packets destined for Customer B's network from the Internet).

Figure 18-5: VLAN Packet Counters - Edge Switch Scenario



Non edge switch scenario

This network is shown in **Figure 18-6 on page 18.27**. The total data count is the upload count plus the download packet count.

Customer A data count

The upload data count for customer A is determined by monitoring the inbound VLAN 10 packets on ports 1.0.18 and 1.0.20, 1.0.22, and 1.0.24 on switch Y (i.e. the traffic from customer A's network).

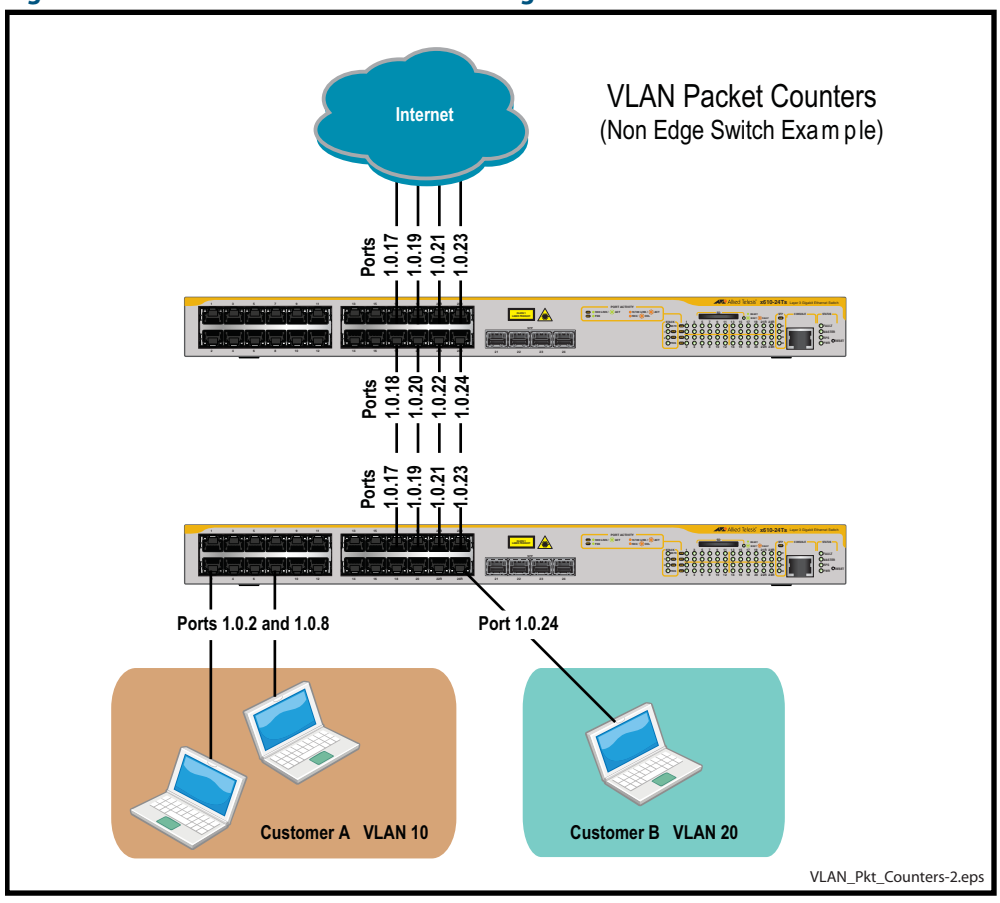
The download data count for customer A is determined by monitoring inbound VLAN 10 packets on ports 1.0.17, 1.0.19, 1.0.21, and 1.0.23.

Customer B data count

The upload data count for customer B is determined by monitoring the inbound VLAN 20 packets on ports 1.0.18 and 1.0.20, 1.0.22, and 1.0.24 on switch Y (i.e. the traffic from customer B's network).

The download data count for customer B is determined by monitoring inbound VLAN 20 packets on ports 1.0.17, 1.0.19, 1.0.21, and 1.0.23.

Figure 18-6: VLAN Packet Counters - Non Edge Switch Scenario



Chapter 19: VLAN Commands

Command List 19.2
clear vlan statistics 19.2
port-vlan-forwarding-priority 19.3
private-vlan 19.6
private-vlan association 19.7
show port-vlan-forwarding-priority 19.8
show vlan 19.9
show vlan classifier group 19.10
show vlan classifier group interface 19.11
show vlan classifier interface group 19.12
show vlan classifier rule 19.13
show vlan private-vlan 19.14
show vlan statistics 19.15
switchport access vlan 19.16
switchport enable vlan 19.17
switchport mode access 19.18
switchport mode private-vlan 19.19
switchport mode private-vlan trunk promiscuous 19.20
switchport mode private-vlan trunk secondary 19.22
switchport mode trunk 19.24
switchport private-vlan host-association 19.25
switchport private-vlan mapping 19.26
switchport trunk allowed vlan 19.27
switchport trunk native vlan 19.30
switchport vlan-stacking (double tagging) 19.31
switchport voice dscp 19.32
switchport voice vlan 19.33
switchport voice vlan priority 19.35
vlan 19.36
vlan classifier activate 19.37
vlan classifier group 19.38
vlan classifier rule ipv4 ..... 19.39
vlan classifier rule proto 19.40
vlan database 19.43
vlan mode stack-local-vlan 19.44
vlan statistics 19.45

Command List

This chapter provides an alphabetical reference of commands used to configure VLANs. For more information see **Chapter 18, VLAN Introduction**.

clear vlan statistics

This command resets the counters for either a specific VLAN statistics instance or (by not specifying an instance) resets the counters for all instances.

Syntax `clear vlan statistics [name <instance_name>]`

Parameter	Description
`vlan statistics`	The count of incoming frames or bytes collected on a per VLAN basis.[1]
`<instance-name>`	The name of the instance for which incoming frames and their bytes are counted.[1]

The terms **frame** and **packet** are used interchangeably.

Mode Privileged Exec

Examples To reset all packet counters for the packet counter instance **vlan2-data:**

```
awplus# clear vlan statistics name vlan2-data
```

To reset all packet counters for all packet counter instances.

```
awplus# clear vlan statistics
```

Related Commands **show vlan statistics**
vlan statistics

port-vlan-forwarding-priority

Use this command to set the highest priority protocol to control transitions from blocking to forwarding traffic. This command prioritizes switch port forwarding mode control, when more than one of EPSR, Loop Protection, and MAC thrashing protection protocols are used on the switch.

EPSR, Loop Protection and MAC Thrashing use the same mechanism to block or forward traffic. This command sets the highest priority protocol to control transitions from blocking to forwarding traffic. Setting the priority stops contention between protocols.

For example, If EPSR is set to the highest priority protocol to block traffic on vlan10 on port1.0.2 then this stops MAC Thrashing from forwarding traffic on vlan10 on port1.0.2.

Caution **loop-protection** and **none** parameter options must not be set on an EPSR master node. Use the **epsr** parameter option on an EPSR master node instead. Setting this command incorrectly on an EPSR master node could cause unexpected broadcast storms.

Use the **no** variant of this command to restore the default highest priority protocol back to the default of EPSR.

For more information about EPSR, see the section **"EPSR Introduction and Configuration" on page 83.1**.

Syntax

```
port-vlan-forwarding-priority {epsr|loop-protection|none}
no port-vlan-forwarding-priority
```

Parameter	Description
`epsr`	Sets EPSR as the highest priority protocol. Use this parameter on an EPSR master node to avoid unexpected broadcast storms.
`loop-protection`	Sets Loop Protection as the highest priority protocol. Note that this option must not be set on an EPSR master node. Use the **epsr** parameter option on an EPSR master node to avoid unexpected broadcast storms.
`none`	Sets the protocols to have equal priority. This was the previous behavior before this command was added, and allows protocols to override each other to set a port to forwarding a VLAN. Note that this option must not be set on a EPSR master node. Use the **epsr** parameter option on an EPSR master node to avoid unexpected broadcast storms.

Default By default, the highest priority protocol is EPSR

Mode Global Configuration

Usage EPSR, Loop Protection and MAC Thrashing protection do not usually need to be configured on a switch, because they perform similar functions—each prevents network loops by blocking a selected port for each (loop containing) VLAN.

However, if more than one of these three features is configured on a switch, you can use this command to prioritize either EPSR or Loop Protection when their effects on a port would conflict and override each other. Previously, each protocol could set a port to forwarding for a VLAN, sometimes overriding the previous setting by another protocol to block the port. This could sometimes lead to unexpected broadcast storms.

Now, when a protocol is set to have the highest priority over a data VLAN on a port, it will not allow other protocols to put that port-vlan into a forwarding state if the highest priority protocol blocked it.

The priority mechanism is only used for blocking-to-forwarding transitions; protocols remain independent on the forwarding-to-blocking transitions.

For example, with an EPSR master node in a two-node ESPR ring with the below settings:

- The EPSR master node primary port is configured to switchport interface **port1.0.1**
- The EPSR master node secondary port is configured to switchport interface **port1.0.2**
- The EPSR master node control VLAN is configured to VLAN interface **vlan10**
- The EPSR master node has a first data VLAN configured to VLAN interface **vlan20**
- The EPSR master node has a second data VLAN configured to VLAN interface **vlan30**.

Initially, the EPSR ring is complete, with **port1.0.2** blocking data VLANs **vlan20** and **vlan30** and some broadcast traffic flowing through. If the user removes **vlan30** from EPSR, a storm is created on **vlan30**. MAC thrashing protection detects it and blocks **vlan30**.

Then after the storm has stopped, MAC thrashing protection sets it to forwarding again and it keeps oscillating between forwarding and blocking. In the meantime, the user adds back **vlan30** to EPSR as a data VLAN and EPSR blocks it on **port1.0.2**.

If the priority is set to none (**port-vlan-forwarding-priority none**), MAC thrashing protection notices that the storm has stopped again and decides to put **vlan30** on **port1.0.2** into forwarding state. This overrides what EPSR requires for this port-VLAN and creates a storm. This matches the old behavior before this feature was implemented.

If the priority is set to EPSR or default (**port-vlan-forwarding-priority epsr**), MAC thrashing protection notices that the storm has stopped again and attempts to put **vlan30** on **port1.0.2** into forwarding state. The higher priority protocol (EPSR) is blocking the VLAN on this port, so it stays blocking and no storm occurs.

Example To prioritize EPSR over Loop Protection or MAC Thrashing protection settings, so that Loop Protection or MAC Thrashing protection cannot set a port to the forwarding state a VLAN if EPSR has set it to the blocking state, use the commands:

```
awplus# configure terminal
awplus(config)# port-vlan-forwarding-priority epsr
```

To prioritize Loop Protection over EPSR or MAC Thrashing protection settings, so that EPSR or MAC Thrashing protection cannot set a port to the forwarding state a VLAN if Loop Protection has set it to the blocking state, use the commands:

```
awplus# configure terminal
awplus(config)# port-vlan-forwarding-priority loop-
protection
```

To set EPSR, Loop Protection, and MAC Thrashing protection protocols to have equal priority for port forwarding and blocking, which allows the protocols to override each other to set a port to the forwarding or blocking states, use the commands:

```
awplus# configure terminal
awplus(config)# port-vlan-forwarding-priority none
```

To restore the default highest priority protocol back to the default of EPSR, use the commands:

```
awplus# configure terminal
awplus(config)# no port-vlan-forwarding-priority
```

Related Commands **show port-vlan-forwarding-priority**

private-vlan

Use this command to a create a private VLAN. Private VLANs can be either primary or secondary. Secondary VLANs can be ether community or isolated.

Use the **no** variant of this command to remove the specified private VLAN.

For more information, see the section **"Private VLANs" on page 18.11**.

Syntax

```
private-vlan <vlan-id> {community|isolated|primary}
no private-vlan <vlan-id> {community|isolated|primary}
```

Parameter	Description
<vlan-id>	VLAN ID in the range <2-4094> for the VLAN which is to be made a private VLAN.
`community`	Community VLAN.
`isolated`	Isolated VLAN.
`primary`	Primary VLAN.

Mode VLAN Configuration

Examples

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 2 name vlan2 state enable
awplus(config-vlan)# vlan 3 name vlan3 state enable
awplus(config-vlan)# vlan 4 name vlan4 state enable
awplus(config-vlan)# private-vlan 2 primary
awplus(config-vlan)# private-vlan 3 isolated
awplus(config-vlan)# private-vlan 4 community
```

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# no private-vlan 2 primary
awplus(config-vlan)# no private-vlan 3 isolated
awplus(config-vlan)# no private-vlan 4 community
```

private-vlan association

Use this command to associate a secondary VLAN to a primary VLAN. Only one isolated VLAN can be associated to a primary VLAN. Multiple community VLANs can be associated to a primary VLAN.

Use the **no** variant of this command to remove association of all the secondary VLANs to a primary VLAN.

For more information, see the section **"Private VLANs" on page 18.11**.

Syntax

```
private-vlan <primary-vlan-id> association {add <secondary-vlan-id>|
    remove <secondary-vlan-id>}

no private-vlan <primary-vlan-id> association
```

Parameter	Description
<primary-vlan-id>	VLAN ID of the primary VLAN.
<secondary-vlan-id>	VLAN ID of the secondary VLAN (either isolated or community).

Mode VLAN Configuration

Examples The following commands associate primary VLAN 2 with secondary VLAN 3:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# private-vlan 2 association add 3
```

The following commands remove the association of primary VLAN 2 with secondary VLAN 3:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# private-vlan 2 association remove 3
```

The following commands remove all secondary VLAN associations of primary VLAN 2:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# no private-vlan 2 association
```

show port-vlan-forwarding-priority

Use this command to display the highest priority protocol that controls port-vlan forwarding or blocking traffic. This command displays whether EPSR or Loop Protection is set as the highest priority for determining whether a port forwards a VLAN, as set by the **port-vlan-forwarding-priority** command.

For more information about EPSR, see the section **"EPSR Introduction and Configuration" on page 83.1**.

Syntax

```
show port-vlan-forwarding-priority
```

Mode

Privileged Exec

Example

To display the highest priority protocol, use the command:

```
awplus# show port-vlan-forwarding-priority
```

Output

Figure 19-1: Example output from the show port-vlan-forwarding-priority command

```
Port-vlan Forwarding Priority: EPSR
```

Related Commands

port-vlan-forwarding-priority

show vlan

Use this command to display information about a particular VLAN by specifying the VLAN ID. It displays information for all the VLANs configured.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show vlan {all|brief|dynamic|static|auto|static-ports<1-4094>}`

Parameter	Description
<1-4094>	Display information about the VLAN specified by the VLAN ID.
`all`	Display information about all VLANs on the device.
`brief`	Display information about all VLANs on the device.
`dynamic`	Display information about all VLANs learned dynamically.
`static`	Display information about all statically configured VLANs.
`auto`	Display information about all auto-configured VLANs.
`static-ports`	Display static egress/forbidden ports.

Mode User Exec and Privileged Exec

Example To display information about VLAN 2, use the command:

```
awplus# show vlan 2
```

Output **Figure 19-2: Example output from the show vlan command**

```
VLAN ID  Name            Type    State   Member ports
                                         (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ===================================
2       VLAN0002         STATIC  ACTIVE  port1.0.3(u) port1.0.4(u) port1.0.5(u)
                                         port1.0.6(u)
.
.
```

Related Commands vlan

show vlan classifier group

Use this command to display information about all configured VLAN classifier groups or a specific group.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show vlan classifier group [<1-16>]`

Parameter	Description
<1-16>	VLAN classifier group identifier

Mode User Exec and Privileged Exec

Usage If a group ID is not specified, all configured VLAN classifier groups are shown. If a group ID is specified, a specific configured VLAN classifier group is shown.

Example To display information about VLAN classifier group 1, enter the command:

```
awplus# show vlan classifier group 1
```

Related Commands **vlan classifier group**

show vlan classifier group interface

Use this command to display information about a single switch port interface for all configured VLAN classifier groups.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show vlan classifier group interface <switch-port>`

Parameter	Description
<switch-port>	Specify the switch port interface classifier group identifier

Mode User Exec and Privileged Exec

Usage All configured VLAN classifier groups are shown for a single interface.

Example To display VLAN classifier group information for switch port interface `port1.0.2`, enter the command:

```
awplus# show vlan classifier group interface port1.0.2
```

Output **Figure 19-3: Example output from the show vlan classifier group interface port1.0.1 command:**

```
vlan classifier group 1 interface port1.0.1
```

Related Commands **vlan classifier group**
show vlan classifier interface group

show vlan classifier interface group

Use this command to display information about all interfaces configured for a VLAN group or all the groups.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show vlan classifier interface group [<1-16>]`

Parameter	Description
<1-16>	VLAN classifier interface group identifier

Mode User Exec and Privileged Exec

Usage If a group ID is not specified, all interfaces configured for all VLAN classifier groups are shown. If a group ID is specified, the interfaces configured for this VLAN classifier group are shown.

Example To display information about all interfaces configured for all VLAN groups, enter the command:

```
awplus# show vlan classifier interface group
```

To display information about all interfaces configured for VLAN group 1, enter the command:

```
awplus# show vlan classifier interface group 1
```

Output **Figure 19-4: Example output from the show vlan classifier interface group command**

```
vlan classifier group 1 interface port1.0.1
vlan classifier group 1 interface port1.0.2
vlan classifier group 2 interface port1.0.3
vlan classifier group 2 interface port1.0.4
```

Output **Figure 19-5: Example output from the show vlan classifier interface group 1 command**

```
vlan classifier group 1 interface port1.0.1
vlan classifier group 1 interface port1.0.2
```

Related Commands **vlan classifier group**
show vlan classifier group interface

show vlan classifier rule

Use this command to display information about all configured VLAN classifier rules or a specific rule.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show vlan classifier rule [<1-256>]`

Parameter	Description
<1-256>	VLAN classifier rule identifier

Mode User Exec and Privileged Exec

Usage If a rule ID is not specified, all configured VLAN classifier rules are shown. If a rule ID is specified, a specific configured VLAN classifier rule is shown.

Example To display information about VLAN classifier rule 1, enter the command:

```
awplus# show vlan classifier rule 1
```

Output **Figure 19-6: Example output from the show vlan classifier rule 1 command**

```
vlan classifier group 1 add rule 1
```

Related Commands
vlan classifier activate
vlan classifier rule ipv4
vlan classifier rule proto

show vlan private-vlan

Use this command to display the private VLAN configuration and associations.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show vlan private-vlan
```

Mode

User Exec and Privileged Exec

Example

To display the private VLAN configuration and associations, enter the command:

```
awplus# show vlan private-vlan
```

Output

Figure 19-7: Example output from the show vlan private-vlan command

```
awplus#show vlan private-vlan
 PRIMARY       SECONDARY          TYPE          INTERFACES
 -------       ---------        ----------      ----------
       2               3          isolated
       2               4         community
                       8          isolated
```

Related Commands

private-vlan
private-vlan association

show vlan statistics

Use this command to display the current configuration for either a specific VLAN statistics instance, or (by not specifying an instance) display all VLAN packet counter instances.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show vlan statistics [name <instance_name>]`

Mode User Exec and Privileged Exec

Examples To display all packet counters for the packet counter instance **vlan2-data**

```
awplus# show vlan statistics name vlan2-data
```

To display all packet counters for all packet counter instances.

```
awplus# show vlan statistics
```

Figure 19-8: Example output from the show vlan statistics command

```
VLAN Stats Collection: vlan2-data
  VLAN ID: 2
  Port Map: port1.0.1, port1.0.2, port1.0.4
  Ingress Packets: total 941, bytes 66185
```

Related Commands **clear vlan statistics**
vlan statistics

switchport access vlan

Use this command to change the port-based VLAN of the current port.

Use the **no** variant of this command to change the port-based VLAN of this port to the default VLAN, *vlan1*.

Syntax

```
switchport access vlan <vlan-id>
no switchport access vlan
```

Parameter	Description
<vlan-id>	<1-4094> The port-based VLAN ID for the port.

Default Reset the default VLAN 1 to specified switchports using the negated form of this command.

Mode Interface Configuration

Usage Any untagged frame received on this port will be associated with the specified VLAN.

Examples To change the port-based VLAN to VLAN 3 for `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport access vlan 3
```

To reset the port-based VLAN to the default VLAN 1 for `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no switchport access vlan
```

Validation Command **show interface switchport**

Related Commands **show vlan**

switchport enable vlan

This command enables the VLAN on the port manually once disabled by certain actions, such as QSP (QoS Storm Protection) or EPSR (Ethernet Protection Switching Ring). Note that if the VID is not given, all disabled VLANs are re-enabled.

Syntax `switchport enable vlan [<1-4094>]`

Parameter	Description
`vlan`	Re-enables the VLAN on the port.
<1-4094>	VLAN ID.

Mode Interface Configuration

Example To re-enable the `port1.0.1` from VLAN 1:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# switchport enable vlan 1
```

Related Commands **show mls qos interface storm-status**
storm-window

switchport mode access

Use this command to set the switching characteristics of the port to access mode. Received frames are classified based on the VLAN characteristics, then accepted or discarded based on the specified filtering criteria.

Syntax

```
switchport mode access [ingress-filter {enable|disable}]
```

Parameter	Description
`ingress-filter`	Set the ingress filtering for the received frames.
`enable`	Turn on ingress filtering for received frames. This is the default.
`disable`	Turn off ingress filtering to accept frames that do not meet the classification criteria.

Default By default, ports are in access mode with ingress filtering on.

Usage Use access mode to send untagged frames only.

Mode Interface Configuration

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode access ingress-filter
enable
```

Validation Command **show interface switchport**

switchport mode private-vlan

Use this command to make a Layer 2 port a private VLAN host port or a promiscuous port.

Use the **no** variant of this command to remove the configuration.

Syntax

```
switchport mode private-vlan {host|promiscuous}
no switchport mode private-vlan {host|promiscuous}
```

Parameter	Description
`host`	This port type can communicate with all other host ports assigned to the same community VLAN, but it cannot communicate with the ports in the same isolated VLAN. All communications outside of this VLAN must pass through a promiscuous port in the associated primary VLAN.
`promiscuous`	A promiscuous port can communicate with all interfaces, including the community and isolated ports within a private VLAN.

Mode Interface Configuration

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode private-vlan host
awplus(config)# interface port1.0.3
awplus(config-if)# switchport mode private-vlan promiscuous
awplus(config)# interface port1.0.4
awplus(config-if)# no switchport mode private-vlan promiscuous
```

Related Commands switchport private-vlan mapping

switchport mode private-vlan trunk promiscuous

Use this command to enable a port in trunk mode to be promiscuous port for isolated VLANs.

Note

Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. Private VLAN trunk ports and GVRP are mutually exclusive.

Use the **no** variant of this command to remove a port in trunk mode as a promiscuous port for isolated VLANs. You must first remove the secondary port, or ports, in trunk mode associated with the promiscuous port with the **no switchport mode private-vlan trunk secondary** command.

Syntax

```
switchport mode private-vlan trunk promiscuous group <group-id>
no switchport mode private-vlan trunk promiscuous
```

Parameter	Description
<group-id>	The group ID is a numeric value in the range 1 to 32 that is used to associate the promiscuous port with secondary ports.

Default By default, a port in trunk mode is disabled as a promiscuous port.

Mode Interface Configuration

Usage A port must be put in trunk mode with **switchport mode trunk** command before it can be enabled as a promiscuous port.

To add VLANs to be trunked over the promiscuous port, use the **switchport trunk allowed vlan** command. These VLANs can be isolated VLANs, or non-private VLANs.

To configure the native VLAN for the promiscuous port, use the **switchport trunk native vlan** command. The native VLAN can be an isolated VLAN, or a non-private VLAN.

When you enable a promiscuous port, all of the secondary port VLANs associated with the promiscuous port via the group ID number must be added to the promiscuous port. In other words, the set of VLANs on the promiscuous port must be a superset of all the VLANs on the secondary ports within the group.

For further information, see **"Private VLANs for Trunked Ports" on page 18.17**.

Examples To create the isolated VLANs 2, 3 and 4 and then enable `port1.0.2` in trunk mode as a promiscuous port for these VLANs with the group ID of 3, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 2-4
awplus(config-vlan)# private-vlan 2 isolated
awplus(config-vlan)# private-vlan 3 isolated
awplus(config-vlan)# private-vlan 4 isolated
awplus(config-vlan)# exit
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode trunk
awplus(config-if)# switchport trunk allowed vlan add 2-4
awplus(config-if)# switchport mode private-vlan trunk
promiscuous group 3
```

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no switchport mode private-vlan trunk
promiscuous
```

Note that you must remove the secondary port or ports enabled as trunk ports that are associated with the promiscuous port before removing the promiscuous port.

Related Commands
switchport mode private-vlan trunk secondary
switchport mode trunk
switchport trunk allowed vlan
switchport trunk native vlan
show vlan private-vlan

switchport mode private-vlan trunk secondary

Use this command to enable a port in trunk mode to be a secondary port for isolated VLANs.

Note Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. Private VLAN trunk ports and GVRP are mutually exclusive.

Use the **no** variant of this command to remove a port in trunk mode as a secondary port for isolated VLANs.

Syntax

```
switchport mode private-vlan trunk secondary group <group-id>
no switchport mode private-vlan trunk secondary
```

Parameter	Description
<group-id>	The group ID is a numeric value in the range 1 to 32 that is used to associate a secondary port with its promiscuous port.

Default By default, a port in trunk mode is disabled as a secondary port.

When a port in trunk mode is enabled to be a secondary port for isolated VLANs, by default it will have a native VLAN of **none** (no native VLAN specified).

Mode Interface Configuration

Usage A port must be put in trunk mode with **switchport mode trunk** command before the port is enabled as a secondary port in trunk mode.

To add VLANs to be trunked over the secondary port use the **switchport trunk allowed vlan** command. These must be isolated VLANs and must exist on the associated promiscuous port.

To configure the native VLAN for the secondary port, use the **switchport trunk native vlan** command. The native VLAN must be an isolated VLAN and must exist on the associated promiscuous port.

For further information, see **"Private VLANs for Trunked Ports" on page 18.17**.

Examples To create isolated private VLAN 2 and then enable `port1.0.3` in trunk mode as a secondary port for the this VLAN with the group ID of `3`, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 2
awplus(config-vlan)# private-vlan 2 isolated
awplus(config-vlan)# exit
awplus(config)# interface port1.0.3
awplus(config-if)# switchport mode trunk
awplus(config-if)# switchport trunk allowed vlan add 2
awplus(config-if)# switchport mode private-vlan trunk
secondary group 3
```

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# no switchport mode private-vlan trunk
secondary
```

Related Commands
switchport mode private-vlan trunk promiscuous
switchport mode trunk
switchport trunk allowed vlan
switchport trunk native vlan
show vlan private-vlan

switchport mode trunk

Use this command to set the switching characteristics of the port to trunk. Received frames are classified based on the VLAN characteristics, then accepted or discarded based on the specified filtering criteria.

Syntax

```
switchport mode trunk [ingress-filter {enable|disable}]
```

Parameter	Description
`ingress-filter`	Set the ingress filtering for the frames received.
`enable`	Turn on ingress filtering for received frames. This is the default.
`disable`	Turn off ingress filtering to accept frames that do not meet the classification criteria.

Default By default, ports are in access mode, are untagged members of the default VLAN (vlan1), and have ingress filtering on.

Mode Interface Configuration

Usage A port in trunk mode can be a tagged member of multiple VLANs, and an untagged member of one native VLAN.

To configure which VLANs this port will trunk for, use the **switchport trunk allowed vlan** command.

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# switchport mode trunk ingress-filter enable
```

Validation Command **show interface switchport**

switchport private-vlan host-association

Use this command to associate a primary VLAN and a secondary VLAN to a host port. Only one primary and secondary VLAN can be associated to a host port.

Use the **no** variant of this command to remove the association.

Syntax

```
switchport private-vlan host-association <primary-vlan-id> add
   <secondary-vlan-id>
no switchport private-vlan host-association
```

Parameter	Description
<primary-vlan-id>	VLAN ID of the primary VLAN.
<secondary-vlan-id>	VLAN ID of the secondary VLAN (either isolated or community).

Mode Interface Configuration

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport private-vlan host-association 2
add 3

awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no switchport private-vlan host-association
```

switchport private-vlan mapping

Use this command to associate a primary VLAN and a set of secondary VLANs to a promiscuous port.

Use the **no** variant of this to remove all the association of secondary VLANs to primary VLANs for a promiscuous port.

Syntax

```
switchport private-vlan mapping <primary-vlan-id> add
   <secondary-vid-list>

switchport private-vlan mapping <primary-vlan-id> remove
   <secondary-vid-list>

no switchport private-vlan mapping
```

Parameter	Description
<primary-vlan-id>	VLAN ID of the primary VLAN.
<secondary-vid-list>	VLAN ID of the secondary VLAN (either isolated or community), or a range of VLANs, or a comma-separated list of VLANs and ranges.

Mode Interface Configuration

Usage This command can be applied to a switch port or a static channel group, but not a dynamic (LACP) channel group. LACP channel groups (dynamic/LACP aggregators) cannot be promiscuous ports in private VLANs.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport private-vlan mapping 2 add 3-4
awplus(config-if)# switchport private-vlan mapping 2 remove 3-4
awplus(config-if)# no switchport private-vlan mapping
```

Related Commands switchport mode private-vlan

switchport trunk allowed vlan

Use this command to add VLANs to be trunked over this switch port. Traffic for these VLANs can be sent and received on the port.

Use the **no** variant of this command to reset switching characteristics of a specified interface to negate a trunked configuration specified with **switchport trunk allowed vlan** command.

Syntax

```
switchport trunk allowed vlan all
switchport trunk allowed vlan none
switchport trunk allowed vlan add <vid-list>
switchport trunk allowed vlan remove <vid-list>
switchport trunk allowed vlan except <vid-list>
no switchport trunk
```

Parameter	Description
`all`	Allow all VLANs to transmit and receive through the port.
`none`	Allow no VLANs to transmit and receive through the port.
`add`	Add a VLAN to transmit and receive through the port. Only use this parameter if a list of VLANs are already configured on a port.
`remove`	Remove a VLAN from transmit and receive through the port. Only use this parameter if a list of VLANs are already configured on a port.
`except`	All VLANs, except the VLAN for which the VID is specified, are part of its port member set. Only use this parameter to remove VLANs after either this parameter or the **all** parameter have added VLANs to a port.
`<vid-list>`	<2-4094> The ID of the VLAN or VLANs that will be added to, or removed from, the port. A single VLAN, VLAN range, or comma-separated VLAN list can be set. For a VLAN range, specify two VLAN numbers: lowest, then highest number in the range, separated by a hyphen. For a VLAN list, specify the VLAN numbers separated by commas. Do not enter spaces between hyphens or commas when setting parameters for VLAN ranges or lists.

Default By default, ports are untagged members of the default VLAN (vlan1).

Mode Interface Configuration

Usage The **all** parameter sets the port to be a tagged member of all the VLANs configured on the device. The **none** parameter removes all VLANs from the port's tagged member set. The **add** and **remove** parameters will add and remove VLANs to and from the port's member set. See the note below about restrictions when using the **add**, **remove**, **except**, and **all** parameters.

Note: Only use the **add** or the **remove** parameters with this command if a list of VLANs are configured on a port. Only use the **except** parameter to remove VLANs after either the **except** or the **all** parameters have first been used to add a list of VLANs to a port.

To remove a VLAN, where the configuration for `port1.0.6` shows the below output:

```
awplus#show running-config
!
interface port1.0.6
switchport
switchport mode trunk
switchport trunk allowed vlan except 4
```

Remove VLAN `3` by re-entering the **except** parameter with the list of VLANs to remove, instead of using the **remove** parameter, as shown in the command example below:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# switchport trunk allowed vlan except 3,4
```

Then the configuration is changed after entering the above commands to remove VLAN 3:

```
awplus#show running-config
!
interface port1.0.6
switchport
switchport mode trunk
switchport trunk allowed vlan except 3-4
```

To add a VLAN, where the configuration for `port1.0.6` shows the below output:

```
awplus#show running-config
!
interface port1.0.6
switchport
switchport mode trunk
switchport trunk allowed vlan except 3-5
```

Add VLAN `4` by re-entering the **except** parameter with a list of VLANs to exclude, instead of using the **add** parameter to include VLAN 4, as shown in the command example below:

```
awplus# configure terminal
awplus(config)# interface port1.0.5
awplus(config-if)# switchport trunk allowed vlan except 3,5
```

The configuration is changed after entering the above commands to add VLAN 4:

```
awplus#show running-config
!
interface port1.0.5
switchport
switchport mode trunk
switchport trunk allowed vlan except 3,5
```

Examples The following shows adding a single VLAN to the port's member set.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk allowed vlan add 2
```

The following shows adding a range of VLANs to the port's member set.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk allowed vlan add 2-4
```

The following shows adding a list of VLANs to the port's member set.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk allowed vlan add 2,3,4
```

switchport trunk native vlan

Use this command to configure the native VLAN for this port. The native VLAN is used for classifying the incoming untagged packets. Use the **none** parameter with this command to remove the native VLAN from the port and set the acceptable frame types to vlan-tagged only.

Use the **no** variant of this command to revert the native VLAN to the default VLAN ID 1. Command negation removes tagged VLANs, and sets the native VLAN to the default VLAN.

Syntax

```
switchport trunk native vlan {<vid>|none}
no switchport trunk native vlan
```

Parameter	Description
`<vid>`	`<2-4094>` The ID of the VLAN that will be used to classify the incoming untagged packets. The VLAN ID must be a part of the VLAN member set of the port.
`none`	No native VLAN specified. This option removes the native VLAN from the port and sets the acceptable frame types to vlan-tagged only. Note: Use the **no** variant of this command to revert to the default VLAN 1 as the native VLAN for the specified interface switchport - not **none**.

Default VLAN 1 (the default VLAN), which is reverted to using the **no** form of this command.

Mode Interface Configuration

Examples The following commands show configuration of VLAN 2 as the native VLAN for interface `port1.0.2`:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk native vlan 2
```

The following commands show the removal of the native VLAN for interface `port1.0.2`:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport trunk native vlan none
```

The following commands revert the native VLAN to the default VLAN 1 for interface `port1.0.2`:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no switchport trunk native vlan
```

switchport vlan-stacking (double tagging)

Use this command to enable VLAN stacking on a port and set it to be a customer-edge-port or provider-port. This is sometimes referred to as VLAN double-tagging, nested VLANs, or QinQ.

Use **no** parameter with this command to disable VLAN stacking on an interface.

Syntax

```
switchport vlan-stacking {customer-edge-port|provider-port}
no switchport vlan-stacking
```

Parameter	Description
`customer-edge-port`	Set the port to be a customer edge port. This port must already be in access mode.
`provider-port`	Set the port to be a provider port. This port must already be in trunk mode.

Default By default, ports are not VLAN stacking ports.

Mode Interface Configuration

Usage Use VLAN stacking to separate traffic from different customers to that they can be managed over a provider network

Traffic with an extra VLAN header added by VLAN stacking cannot be routed.

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport vlan-stacking customer-edge-port
```

switchport voice dscp

Use this command to configure the Layer 3 DSCP value advertised when the transmission of LLDP-MED Network Policy TLVs for voice devices is enabled. When LLDP-MED capable IP phones receive this network policy information, they transmit voice data with the specified DSCP value.

Use the **no** variant of this command to reset the DSCP value to the default, 0.

Syntax

```
switchport voice dscp <0-63>
no switchport voice dscp
```

Parameter	Description
`dscp`	Specify a DSCP value for voice data.
<0-63>	DSCP value.

Default A DSCP value of 0 will be advertised.

Mode Interface Configuration

Usage LLDP-MED advertisements including Network Policy TLVs are transmitted via a port if:

- LLDP is enabled (**lldp run** command on page 97.16)
- Voice VLAN is configured for the port (**switchport voice vlan** command on page 19.33)
- The port is configured to transmit LLDP advertisements—enabled by default (**lldp transmit receive** command on page 97.20)
- The port is configured to transmit Network Policy TLVs—enabled by default (**lldp med-tlv-select** command on page 97.9)
- There is an LLDP-MED device connected to the port

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.5
awplus(config-if)# switchport voice dscp 27
```

Related Commands

lldp med-tlv-select
show lldp
switchport voice vlan

switchport voice vlan

Use this command to configure the Voice VLAN tagging advertised when the transmission of LLDP-MED Network Policy TLVs for voice endpoint devices is enabled. When LLDP-MED capable IP phones receive this network policy information, they transmit voice data with the specified tagging. This command also sets the ports to be spanning tree edge ports, that is, it enables spanning tree portfast on the ports.

Use the **no** variant of this command to remove LLDP-MED network policy configuration for voice devices connected to these ports. This does not change the spanning tree edge port status.

Syntax

```
switchport voice vlan [<vid>|dot1p|dynamic|untagged]
no switchport voice vlan
```

Parameter	Description
`<vid>`	VLAN identifier, in the range 1 to 4094.
`dot1p`	The IP phone should send User Priority tagged packets, that is, packets in which the tag contains a User Priority value, and a VID of 0. (The User Priority tag is also known as the 802.1p priority tag, or the Class of Service (CoS) tag.)
`dynamic`	The VLAN ID with which the IP phone should send tagged packets will be assigned by RADIUS authentication.
`untagged`	The IP phone should send untagged packets.

Default By default, no Voice VLAN is configured, and therefore no network policy is advertised for voice devices.

Mode Interface Configuration

Usage LLDP-MED advertisements including Network Policy TLVs are transmitted via a port if:

- LLDP is enabled (**lldp run** command on page 97.16)
- Voice VLAN is configured for the port using this command (**switchport voice vlan**)
- The port is configured to transmit LLDP advertisements—enabled by default (**lldp transmit receive** command on page 97.20)
- The port is configured to transmit Network Policy TLVs—enabled by default (**lldp med-tlv-select** command on page 97.9)
- There is an LLDP-MED device connected to the port.

To set the priority value to be advertised for tagged frames, use the **switchport voice vlan priority** command on page 19.35.

If the Voice VLAN details are to be assigned by RADIUS, then the RADIUS server must be configured to send the attribute “Egress-VLANID (56)” or “Egress-VLAN-Name (58)” in the RADIUS Accept message when authenticating a phone attached to this port.

To set these attributes on the local RADIUS server, use the **egress-vlan-id** command on page 73.19 or the **egress-vlan-name** command on page 73.20.

For more information about configuring authentication for Voice VLAN, **“Configuring LLDP” on page 96.11**.

If the ports have been set to be edge ports by the **switchport voice vlan** command, the **no** variant of this command will leave them unchanged as edge ports. To set them back to their default non-edge port configuration, use the **spanning-tree edgeport (RSTP and MSTP)** command on page 21.35.

Examples To tell IP phones connected to `port1.0.5` to send voice data tagged for VLAN 10, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.5
awplus(config-if)# switchport voice vlan 10
```

To tell IP phones connected to ports 1.0.2-1.0.6 to send priority tagged packets (802.1p priority tagged with VID 0, so that they will be assigned to the port VLAN) use the following commands. The priority value is 5 by default, but can be configured with the **switchport voice vlan priority** command.

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.6
awplus(config-if)# switchport voice vlan dot1p
```

To dynamically configure the VLAN ID advertised to IP phones connected to `port1.0.1` based on the VLAN assigned by RADIUS authentication (with RADIUS attribute “Egress-VLANID” or “Egress-VLAN-Name” in the RADIUS accept packet), use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# switchport voice vlan dynamic
```

To remove the Voice VLAN, and therefore disable the transmission of LLDP-MED network policy information for voice devices on `port1.0.6`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# no switchport voice vlan
```

Related Commands
egress-vlan-id
egress-vlan-name
lldp med-tlv-select
spanning-tree edgeport (RSTP and MSTP)
switchport voice dscp
switchport voice vlan priority
show lldp

switchport voice vlan priority

Use this command to configure the Layer 2 user priority advertised when the transmission of LLDP-MED Network Policy TLVs for voice devices is enabled. This is the priority in the User Priority field of the IEEE 802.1Q VLAN tag, also known as the Class of Service (CoS), or 802.1p priority. When LLDP-MED capable IP phones receive this network policy information, they transmit voice data with the specified priority.

Syntax

```
switchport voice vlan priority <0-7>
no switchport voice vlan priority
```

Parameter	Description
`priority`	Specify a user priority value for voice data.
<0-7>	Priority value.

Default By default, the Voice VLAN user priority value is 5.

Mode Interface Configuration

Usage LLDP-MED advertisements including Network Policy TLVs are transmitted via a port if:

- LLDP is enabled (**lldp run** command on page 97.16)
- Voice VLAN is configured for the port (**switchport voice vlan** command on page 19.33)
- The port is configured to transmit LLDP advertisements—enabled by default (**lldp transmit receive** command on page 97.20)
- The port is configured to transmit Network Policy TLVs—enabled by default (**lldp med-tlv-select** command on page 97.9)
- There is an LLDP-MED device connected to the port.

To set the Voice VLAN tagging to be advertised, use the **switchport voice vlan** command on page 19.33.

Example To remove the Voice VLAN, and therefore disable the transmission of LLDP-MED network policy information for voice devices on `port1.0.6`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# no switchport voice vlan
```

Related Commands

lldp med-tlv-select
show lldp
switchport voice vlan

vlan

This command creates VLANs, assigns names to them, and enables or disables them. Specifying the `disable` state causes all forwarding over the specified VLAN ID to cease. Specifying the `enable` state allows forwarding of frames on the specified VLAN.

The **no** variant of this command destroys the specified VLANs.

Syntax

```
vlan <vid> [name <vlan-name>] [state {enable|disable}]
vlan <vid-range> [state {enable|disable}]
vlan {<vid>|<vlan-name>} [mtu <mtu-value>]
no vlan {<vid>|<vid-range>} [mtu]
```

Parameter	Description
<vid>	The VID of the VLAN to enable or disable in the range <**1-4094**>.
<vlan-name>	The ASCII name of the VLAN. Maximum length: **32** characters.
<vid-range>	Specifies a range of VLAN identifiers.
<mtu-value>	Specifies the Maximum Transmission Unit (MTU) size in bytes, in the range 68 to1500 bytes, for the VLAN.
`enable`	Sets VLAN into an `enable` state.
`disable`	Sets VLAN into a `disable` state.

Default By default, VLANs are enabled when they are created.

Mode VLAN Configuration

Examples

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 45 name accounts state enable
```

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# no vlan 45
```

Related Commands

mtu
vlan database
show vlan

vlan classifier activate

Use this command in Interface Configuration mode to associate a VLAN classifier group with the switch port.

Use the **no** variant of this command to remove the VLAN classifier group from the switch port.

Syntax

```
vlan classifier activate <vlan-class-group-id>
no vlan classifier activate <vlan-class-group-id>
```

Parameter	Description
<vlan-class-group-id>	Specify a VLAN classifier group identifier in the range <1-16>.

Mode Interface Configuration mode for a switch port.

Usage See the **Protocol Based VLAN Configuration Example** section in **Chapter 18, VLAN Introduction** for a configuration example and network topology using this command.

Example To associate VLAN classifier group 3 with switch `port1.0.3`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# vlan classifier activate 3
```

To remove VLAN classifier group 3 from switch `port1.0.3`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# no vlan classifier activate 3
```

Related Commands

show vlan classifier rule
vlan classifier group
vlan classifier rule ipv4
vlan classifier rule proto

vlan classifier group

Use this command to create a group of VLAN classifier rules. The rules must already have been created.

Use the **no** variant of this command to delete a group of VLAN classifier rules.

Syntax

```
vlan classifier group <1-16> {add|delete} rule <vlan-class-rule-id>
no vlan classifier group <1-16>
```

Parameter	Description
<1-16>	VLAN classifier group identifier
add	Add the rule to the group.
delete	Delete the rule from the group.
<vlan-class-rule-id>	The VLAN classifier rule identifier.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# vlan classifier group 3 add rule 5
```

Related Commands

show vlan classifier rule
vlan classifier activate
vlan classifier rule ipv4
vlan classifier rule proto

vlan classifier rule ipv4

Use this command to create an IPv4 subnet-based VLAN classifier rule and map it to a specific VLAN. Use the **no** variant of this command to delete the VLAN classifier rule.

Syntax

```
vlan classifier rule <1-256> ipv4 <ip-addr/prefix-length>  vlan
   <1-4094>

no vlan classifier rule <1-256>
```

Parameter	Description
<1-256>	Specify the VLAN Classifier Rule identifier.
<ip-addr/prefix-length>	Specify the IP address and prefix length.
<1-4094>	Specify a VLAN ID to which an untagged packet is mapped in the range *<1-4094>*.

Mode Global Configuration

Usage If the source IP address matches the IP subnet specified in the VLAN classifier rule, the received packets are mapped to the specified VLAN.

Example

```
awplus# configure terminal
awplus(config)# vlan classifier rule 3 ipv4 3.3.3.3/8 vlan 5
```

Related Commands

show vlan classifier rule
vlan classifier activate
vlan classifier rule proto

vlan classifier rule proto

Use this command to create a protocol type-based VLAN classifier rule, and map it to a specific VLAN. See the published IANA EtherType IEEE 802 numbers here:

www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.txt.

Instead of a protocol name the decimal value of the protocol's EtherType can be entered. The EtherType field is a two-octet field in an Ethernet frame. It is used to show which protocol is encapsulated in the payload of the Ethernet frame. Note that EtherTypes in the IANA 802 numbers are given as hexadecimal values.

The **no** variant of this command removes a previously set rule.

Syntax

```
vlan classifier rule <1-256> proto <protocol> encap {ethv2|nosnapllc|
    snapllc} vlan <1-4094>

no vlan classifier rule <1-256>
```

Parameter	Description	
<1-256>	VLAN Classifier identifier	
`proto`	Protocol type	
<protocol>	Specify a protocol either by its decimal number (0-65535) or by one of the following protocol names:	
	`[arp|2054]`	Address Resolution protocol
	`[atalkaarp|33011]`	Appletalk AARP protocol
	`[atalkddp|32923]`	Appletalk DDP protocol
	`[atmmulti|34892]`	MultiProtocol Over ATM protocol
	`[atmtransport|34948]`	Frame-based ATM Transport protocol
	`[dec|24576]`	DEC Assigned protocol
	`[deccustom|24582]`	DEC Customer use protocol
	`[decdiagnostics|24581]`	DEC Systems Comms Arch protocol
	`[decdnadumpload|24577]`	DEC DNA Dump/Load protocol
	`[decdnaremoteconsole|24578]`	DEC DNA Remote Console protocol
	`[decdnarouting|24579]`	DEC DNA Routing protocol
	`[declat|24580]`	DEC LAT protocol
	`[decsyscomm|24583]`	DEC Systems Comms Arch protocol

Parameter (cont.)	Description (cont.)	
	`[g8bpqx25|2303]`	G8BPQ AX.25 protocol
	`[ieeeaddrtrans|2561]`	Xerox IEEE802.3 PUP Address
	`[ieeepup|2560]`	Xerox IEEE802.3 PUP protocol
	`[ip|2048]`	IP protocol
	`[ipv6|34525]`	IPv6 protocol
	`[ipx|33079]`	IPX protocol
	`[netbeui|61680]`	IBM NETBIOS/NETBEUI protocol
	`[netbeui|61681]`	IBM NETBIOS/NETBEUI protocol
	`[pppdiscovery|34915]`	PPPoE discovery protocol
	`[pppsession|34916]`	PPPoE session protocol
	`[rarp|32821]`	Reverse Address Resolution protocol
	`[x25|2056]`	CCITT.25 protocol
	`[xeroxaddrtrans|513]`	Xerox PUP Address Translation protocol
	`[xeroxpup|512]`	Xerox PUP protocol
`ethv2`	Ethernet Version 2 encapsulation	
`nosnapllc`	LLC without SNAP encapsulation	
`snapllc`	LLC SNAP encapsulation	
`<1-4094>`	Specify a VLAN ID to which an untagged packet is mapped in the range *`<1-4094>`*	

Mode Global Configuration

Usage If the protocol type matches the protocol specified in the VLAN classifier rule, the received packets are mapped to the specified VLAN. Ethernet Frame Numbers may be entered in place of the protocol names listed. For a full list please refer to the IANA list online: **www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.txt**

Examples

```
awplus# configure terminal
awplus(config)# vlan classifier rule 1 proto x25 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 2 proto 512 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 3 proto 2056 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 4 proto 2054 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 5 proto 34525 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 6 proto ipv6 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 7 proto 2048 encap ethv2
vlan 2
awplus(config)# vlan classifier rule 8 proto ip encap ethv2
vlan 2
```

Validation Output

```
awplus# show vlan classifier rule
```

```
vlan classifier rule 16 proto rarp encap ethv2 vlan 2
vlan classifier rule 8 proto encap ethv2 vlan 2
vlan classifier rule 4 proto arp encap ethv2 vlan 2
vlan classifier rule 3 proto xeroxpup encap ethv2 vlan 2
vlan classifier rule 2 proto ip encap ethv2 vlan 2
vlan classifier rule 1 proto ipv6 encap ethv2 vlan 2
```

Related Commands

show vlan classifier rule
vlan classifier activate
vlan classifier group

vlan database

Use this command to enter the VLAN Configuration mode.

Syntax `vlan database`

Mode Global Configuration

Usage Use this command to enter the VLAN configuration mode. You can then add or delete a VLAN, or modify its values.

Example In the following example, note the change to VLAN configuration mode from Configure mode:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)#
```

Related Commands vlan

vlan mode stack-local-vlan

This command enables you to create stack-local-VLANs and use ICMP to monitor and diagnose issues within specific members of the stack. When a VLAN is added using this method, all its traffic will be trapped to and processed by the CPU of the specific local stack member, rather than the CPU of the stack master.

The **no** variant of this command destroys the specified VLAN.

Syntax

```
vlan <vid> mode stack-local-vlan <member-id>
no vlan <vid>
```

Parameter	Description
<vid>	The VID of the VLAN to be created in the range 2-4094. We recommend that the first stack-local-vlan be assigned the number 4001 for the first stack member, then incremented by one for each stack member. So a stack of four members would be assigned the following VID numbers: stack member one VID 4001 stack member two VID 4002 stack member three VID 4003 stack member four VID 4004
`mode stack-local-vlan`	Specifies that the new VLAN will function as a stack-local-VLAN.
<member-id>	Specifies the new stack member ID. Enter a decimal number in the range 1-8.

Default By default, VLANs are automatically enabled as they are added.

Mode VLAN Configuration

Examples To add a stack-local-VLAN with the VID of 4002 and assign it to stack member 2, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 4002 mode stack-local-vlan 2
```

To remove VLAN 4002, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# no vlan 4002
```

vlan statistics

This command creates a VLAN packet counter instance, and enables you to add one or more ports to a defined counter instance. This command can only be applied to switch ports. You cannot apply it to aggregated links or eth ports.

The **no** variant of this command enables the deletion of VLAN packet counter instances, or for removing one or more ports that are currently mapped to a counter instance. Note that the selected range of ports must all be switch ports.

Note In describing this command, the terms frame and packet are used interchangeably.

Syntax

```
vlan <vid> statistics name <instance_name>
no vlan statistics name <instance_name>
```

Parameter	Description
<vid>	The VID of the VLAN that is associated with *<instance-name>*.
<instance-name>	The name of the instance for which incoming frames and their bytes are counted.

Mode Interface Configuration

Usage A maximum of 128 packet counter instances can be created. When the first instance is configured, the switch will reserve sufficient resources to support 128 packet counter instances. These resources are also shared with other features such as QoS and ACLs. Where the remaining resources are insufficient to support the VLAN Statistics feature the feature will not be enabled, and an error message will display.

Examples Create a VLAN packet counter instance named **vlan2-data**, and apply this to count incoming vlan2 tagged frames on ports 1.0.4 and 1.0.5.

```
awplus# configure terminal
awplus(config)# interface port1.0.4,port1.0.5
awplus(config-if)# vlan 2 statistics name vlan2-data
```

From the previous example, add ports in the range 1.0.2 to 1.0.3 to the VLAN packet counter instance. The **vlan2-data** instance will now count all incoming vlan2 tagged frames on ports within the range 1.0.1 to 1.0.5.

```
awplus(config)# interface port1.0.2-port1.0.3
awplus(config-if)# vlan 2 statistics name vlan2-data
```

To remove `port1.0.5` from the packet counter instance named **vlan2-data.**

```
awplus(config)# interface port1.0.5
awplus(config-if)# no vlan statistics name vlan2-data
```

To remove the remaining ports 1.0.2 to 1.0.4 from the packet counter instance named **vlan2-data.** Note that because there are no ports associated with the **vlan2-data**, this instance will be removed.

```
awplus(config)# interface port1.0.2-port1.0.4
awplus(config-if)# no vlan statistics name vlan2-data
```

Related Commands

clear vlan statistics
show vlan statistics

Chapter 20: Spanning Tree Introduction: STP, RSTP, and MSTP

Introduction 20.2
Overview of Spanning Trees 20.2
- Spanning Tree Operation 20.2
- Spanning Tree Modes 20.4

Spanning Tree Protocol (STP) 20.5
Configuring STP 20.6
Rapid Spanning Tree Protocol (RSTP) 20.8
Configuring RSTP 20.9
Multiple Spanning Tree Protocol (MSTP) 20.11
- Multiple Spanning Tree Instances (MSTI) 20.12
- MSTP Regions 20.13
- Common and Internal Spanning Tree (CIST) 20.15
- MSTP Bridge Protocol Data Units (BPDUs) 20.17

Configuring MSTP 20.19

Introduction

This chapter describes and provides configuration procedures for:

- Spanning Tree Protocol (STP)
- Rapid Spanning Tree Protocol (RSTP)
- Multiple Spanning Tree Protocol (MSTP)

For detailed information about the commands used to configure spanning trees, see **Chapter 21, Spanning Tree Commands**.

Overview of Spanning Trees

The concept of the spanning tree protocol was devised to address broadcast storming. The spanning tree algorithm itself is defined by the IEEE standard 802.1D and its later revisions.

The IEEE Standard 802.1 uses the term “bridge” to define the spanning tree operation, and uses terms such as Bridge Protocol Data Units and Root Bridge when defining spanning tree protocol functions.

When a bridge receives a frame, it reads the source and destination address fields. The bridge then enters the frame’s source address in its forwarding database. In doing this the bridge associates the frame’s source address with the network attached to the port on which the frame was received. The bridge also reads the destination address and if it can find this address in its forwarding database, it forwards the frame to the appropriate port. If the bridge does not recognize the destination address, it forwards the frame out from all its ports except for the one on which the frame was received, and then waits for a reply. This process is known as “flooding”.

A significant problem arises where bridges connect via multiple paths. A frame that arrives with an unknown destination address is flooded over all available paths. The arrival of these frames at another network via different paths and bridges produces major problems. The bridges can become confused about the location of the send and receive devices and begin sending frames in the wrong directions. This process is self-perpetuating and produces a condition known as a broadcast storm, where the increase of circulating frames can eventually overload the network.

Spanning Tree Operation

Where a LAN’s topology results in more than one path existing between bridges, frames transmitted onto the extended LAN circulate in increasing numbers around the loop, decreasing performance and potentially overloading the network. However, multiple paths through the extended LAN are often required in order to provide redundancy and backup in the event of a bridge or link failure.

The spanning tree is created through the exchange of Bridge Protocol Data Units (BPDUs) between the bridges in the LAN. The spanning tree algorithm operates by:

- Automatically computing a loop-free portion of the topology, called a *spanning tree*. The topology is dynamically pruned to the spanning tree by declaring certain ports on a switch to be redundant, and placing them into a ‘blocking’ state.
- Automatically recovering from a switch failure that would partition the extended LAN by reconfiguring the spanning tree to use redundant paths, if available.

The logical tree computed by the spanning tree algorithm has the following properties:

- A single bridge is selected to become the spanning tree's unique *root bridge*. This is the device that advertises the lowest Bridge ID. Each bridge is uniquely identified by its Bridge ID, which comprises the bridge's *root priority* (a spanning tree parameter) followed by its MAC address.
- Each bridge or LAN in the tree, except the root bridge, has a unique parent, known as the *designated bridge*. Each LAN has a single bridge, called the *designated bridge*, that connects it to the next LAN on the path towards the root bridge.
- Each port connecting a bridge to a LAN has an associated *cost*, called the *root path cost*. This is the sum of the costs for each path between the particular bridge port and the root bridge. The designated bridge for a LAN is the one that advertises the lowest *root path cost*. If two bridges on the same LAN have the same lowest root path cost, then the switch with the lowest bridge ID becomes the designated bridge.

The spanning tree computation is a continuous, distributed process to establish and maintain a spanning tree (**Table 20-1**). The basic algorithm is similar for STP, RSTP and MSTP modes.

Table 20-1: Spanning tree process

The spanning tree algorithm ...	By ...
Selects a root bridge	It selects as the root bridge for the spanning tree the device with the (numerically) lowest bridge identifier (that is, the device with lowest root bridge priority value, or if they have the same priority, the bridge with the lowest MAC address).
Selects root ports	On each device, it selects the root port according to: ■ the port with the lowest path cost to the root bridge ■ the port connected to the bridge with the lowest root identifier ■ MSTP and RSTP only: the port with the lowest port priority value ■ the port with the lowest port number
Blocks alternate ports	In order to prevent loops, it blocks alternate ports (discarding state) that provide higher cost paths to the root bridge.
Blocks backup ports	Where a second port connects one switch back to itself, it blocks the backup port that has the highest path cost or port number.
Selects designated ports	All other ports that are not disabled are selected as designated ports and are eventually made active (Forwarding state).
Maintains the spanning tree	If a switch or port fails, the spanning tree configures a new active topology, changing some port states, to reestablish connectivity and block loops. Depending on where the failure occurs, the changes may be widespread (e.g. if the root bridge fails), or local (e.g. if a designated port fails).

The logical spanning tree, sometimes called the *active topology*, includes the root bridge and all designated bridges, meaning all ports that are to be used for communication within the spanning tree. These ports are in the forwarding state. Ports removed from the logical spanning tree are not in the forwarding state. To implement the spanning tree algorithm, devices communicate with one another using the Spanning Tree Protocol.

Spanning Tree Modes

STP can run in one of three modes: STP, RSTP or MSTP. A device running RSTP is compatible with other devices running STP; a device running MSTP is compatible with other devices running RSTP or STP. By default, on a device in MSTP mode each port automatically detects the mode of the device connected to it (MSTP, RSTP or STP), and responds in the appropriate mode by sending messages (BPDUs) in the corresponding format. Ports on a device in RSTP mode can automatically detect and respond to connected devices in RSTP and STP mode. Particular ports can also be forced to only operate in a particular mode (**spanning-tree force-version** command on page 21.40).

STP The Spanning Tree Protocol (STP) is the original protocol defined by IEEE standard 802.1D-1988. It creates a single spanning tree over a network.

STP mode may be useful for supporting applications and protocols whose frames may arrive out of sequence or duplicated, for example NetBeui.

RSTP Rapid Spanning Tree Protocol (RSTP) also creates a single spanning tree over a network. Compared with STP, RSTP provides for more rapid convergence to an active spanning tree topology. RSTP is defined in IEEE standard 802.1D-2004.

By default, the device operates in RSTP mode.

MSTP The Multiple Spanning Tree Protocol (MSTP) addresses the limitations in the previous spanning tree protocols, STP and RSTP, within networks that use multiple VLANs with topologies that employ alternative physical links. It supports multiple spanning tree instances on any given link within a network, and supports large networks by grouping bridges into regions that appear as a single bridge to other devices.

MSTP is defined in IEEE standard 802.1Q-2005. The protocol builds on, and remains compatible with, the previous IEEE standards defining STP and RSTP.

Spanning Tree Protocol (STP)

STP uses the process described in **Table 20-1** to avoid loops.

STP port states

In STP mode, each switch port can be in one of five spanning tree states, and one of two switch states. The state of a switch port is taken into account by STP. The STP port states (**Table 20-2**) affect the behavior of ports whose switch state is enabled.

Table 20-2: STP port states

State	Meaning
DISABLED	STP operations are disabled on the port. The port does not participate in the operation of the Spanning Tree Algorithm and Protocol. The port can still switch if its switch state is enabled.
BLOCKING	The forwarding process discards received frames and does not submit forwarded frames for transmission. This is the “standby” mode.The port does not participate in frame relay.
LISTENING	The port is enabled for receiving frames only. The port is preparing to participate in frame relay. The forwarding process discards received frames and does not submit forwarded frames for transmission.
LEARNING	The port is enabled for receiving frames only, and the Learning Process can add new source address information to the Forwarding Database.
FORWARDING	The normal state for a switch port. The forwarding process and the Spanning Tree entity are enabled for transmit and receive operations on the port.

Configuring STP

By default, RSTP is enabled on all switch ports. This section provides a procedure for configuring STP (**Table 20-3**).

To configure other modes, see **"Configuring RSTP" on page 20.9** or **"Configuring MSTP" on page 20.19**.

Table 20-3: Configuration procedure for STP

Command	Description
Step 1. Configure STP RSTP is enabled by default with default settings on all switch ports to prevent Layer 2 loops in your network.	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `spanning-tree mode stp`	By default, the device is in RSTP mode. Change to STP mode.
`awplus(config)#` `spanning-tree enable`	By default, spanning tree is enabled on all switch ports. If it has been disabled, enable it for STP.
`awplus(config)#` `spanning-tree priority <priority>`	By default, all devices have the same root bridge priority, 32768 (8000 in hexadecimal), so the device with the lowest MAC address becomes the root bridge. If you want the device to be the root bridge, set the root bridge priority to a value lower than 32768. Enter a value in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.
Step 2. Configure Root Guard The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP).	
`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for the switch ports you want to enable Root Guard for.
`awplus(config-if)#` `spanning-tree guard root`	Enable the Guard Root feature for these ports.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `exit`	Return to Privileged Exec mode.

Table 20-3: Configuration procedure for STP(cont.)

Command	Description
Step 3. Check STP configuration	
`awplus#` `show spanning-tree [interface <port-list>]`	Display the spanning tree configuration for the device, and confirm the new root bridge priority (Bridge Priority). Note that the Bridge ID is in a form like this: 80000000cd240331, and that other IDs follow the same pattern. This is made up of: 8000—the devices' root bridge priority in hexadecimal 0000cd240331—the devices' MAC address.

Advanced configuration: For most networks the default settings for path costs will be suitable, however, you can configure them if required (**spanning-tree path-cost**).

Rapid Spanning Tree Protocol (RSTP)

RSTP uses the process described in **Table 20-1** to avoid loops.

A spanning tree running in STP mode can take up to one minute to rebuild after a topology or configuration change. The RSTP algorithm provides for a faster recovery of connectivity following the failure of a bridge, bridge port, or a LAN. RSTP provides rapid recovery by including port roles in the computation of port states, and by allowing neighboring bridges to explicitly acknowledge signals on a point-to-point link that indicate that a port wants to enter the forwarding mode.

In rapid mode, the rapid transition of a port to the forwarding state is possible when the port is considered to be part of a point-to-point link, or when the port is considered to be an *edge* port. An edge port is one that attaches to a LAN that has no other bridges attached.

Table 20-4: RSTP port states

State	Meaning
DISABLED	STP operations are disabled on the port.
DISCARDING	The port does not participate in frame relay. The forwarding process discards received frames and does not submit forwarded frames for transmission.
LEARNING	The port is enabled for receiving frames only, and the learning process can add new source address information to the forwarding database. The port does not forward any frames.
FORWARDING	The normal state for a switch port. The forwarding process and the Spanning Tree entity are enabled for transmit and receive operations on the port.

Configuring RSTP

RSTP is enabled by default with default settings on all switch ports to prevent Layer 2 loops in your network. No further configuration is required if you want to use RSTP with these default settings. For further RSTP configuration, see **Table 20-5** below.

To configure other modes, see **"Configuring MSTP" on page 20.19** or **"Configuring STP" on page 20.6**.

For detailed configuration examples, see the How To Note *How To Configure Basic Switching Functionality*, available from **alliedtelesis.com**.

Table 20-5: Configuration procedure for RSTP

Command	Description
Step 1. Configure RSTP RSTP is enabled by default with default settings on all switch ports to prevent Layer 2 loops in your network. No further configuration is required if you want to use RSTP with these default settings. If you need to restore the device to RSTP after it has been set to another mode, or modify the default RSTP settings, follow the procedure below.	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `spanning-tree mode rstp`	By default, the device is in RSTP mode. If it has been changed to STP or MSTP mode, change it back to RSTP.
`awplus(config)#` `spanning-tree enable`	By default, spanning tree is enabled on all switch ports. If it has been disabled, enable it for RSTP.
`awplus(config)#` `spanning-tree priority <priority>`	By default, all devices have the same root bridge priority, 32768 (8000 in hexadecimal), so the device with the lowest MAC address becomes the root bridge. If you want the device to be the root bridge, set the root bridge priority to a value lower than 32768. Enter a value in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.
Step 2. Configure edge ports If some switch ports are connected to devices that cannot generate BPDUs (such as workstations), you can set particular switch ports as edge ports, or set them to automatically detect whether they are edge ports.	
`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for these switch ports.

`awplus(config-if)#`
`spanning-tree edgeport (RSTP and MSTP)`
or
`awplus(config-if)#`
`spanning-tree autoedge (RSTP and MSTP)`
	Set these ports to be edge ports,
	or
	set these ports to automatically detect whether they are edge ports.

Table 20-5: Configuration procedure for RSTP(cont.)

Command	Description
Step 3. Configure Root Guard	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for the switch ports you want to enable Root Guard for.
`awplus(config-if)#` `spanning-tree guard root`	The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP). Enable the Guard Root feature if required.
Step 4. Configure BPDU Guard	
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `spanning-tree portfast bpdu-guard`	If required, enable the BPDU Guard feature.
`awplus(config)#` `spanning-tree errdisable-timeout enable`	Set a timeout for ports that are disabled due to the BPDU guard feature.
`awplus(config)#` `spanning-tree errdisable-timeout interval`	Specify the time interval after which a port is brought back up when it has been disabled by the BPDU guard feature.
Step 5. Check RSTP configuration	
`awplus(config)#` `exit`	Return to Privileged Exec mode.
`awplus#` `show spanning-tree [interface <port-list>]`	Display the spanning tree configuration for the device, and confirm the new root bridge priority (Bridge Priority). Note that the Bridge ID is in a form like this: 80000000cd240331, and that other IDs follow the same pattern. This is made up of: 8000—the devices' root bridge priority in hexadecimal 0000cd240331—the devices' MAC address.

Advanced configuration: For most networks the default settings for path costs will be suitable, however, you can configure them if required (**spanning-tree path-cost**).

Multiple Spanning Tree Protocol (MSTP)

Conceptually, MSTP views the total bridged network as one that comprises a number of *Multiple Spanning Tree Regions* (MSTRs), where each region can contain up to 64 spanning trees, which operate locally, called *Multiple Spanning Tree Instances* (MSTIs). AlliedWare Plus™ supports up to 15 MSTIs. The regions are linked by the *Common Internal Spanning Tree* (CIST).

MSTP uses BPDUs to exchange information between spanning-tree compatible devices, to prevent loops in each MSTI and also in the CIST, by selecting active and blocked paths. This process is described in **Table 20-1**.

If multiple ports are aggregated together into a dynamic (LACP) or static channel group, then the spanning-tree process is aware of the link aggregation and treats the aggregated ports as a single logical path.

Advantage of MSTP over RSTP

MSTP is similar to RSTP, in that it provides loop resolution and rapid convergence. However, RSTP can keep track of only one spanning-tree. MSTP can track many spanning-trees, referred to as *instances*. MSTP makes it possible to have different forwarding paths for different MST instances. This enables load balancing of network traffic across redundant links, so that all the links in a network can be used by at least one MSTI, and no link is left completely idle. That is to say that no link is unnecessarily shut down by spanning-tree.

Essentially, MSTP is VLAN aware and RSTP is not VLAN aware. MSTP BPDUs and RSTP BPDUs are compatible, so a network can have a mixture of MSTP and RSTP areas.

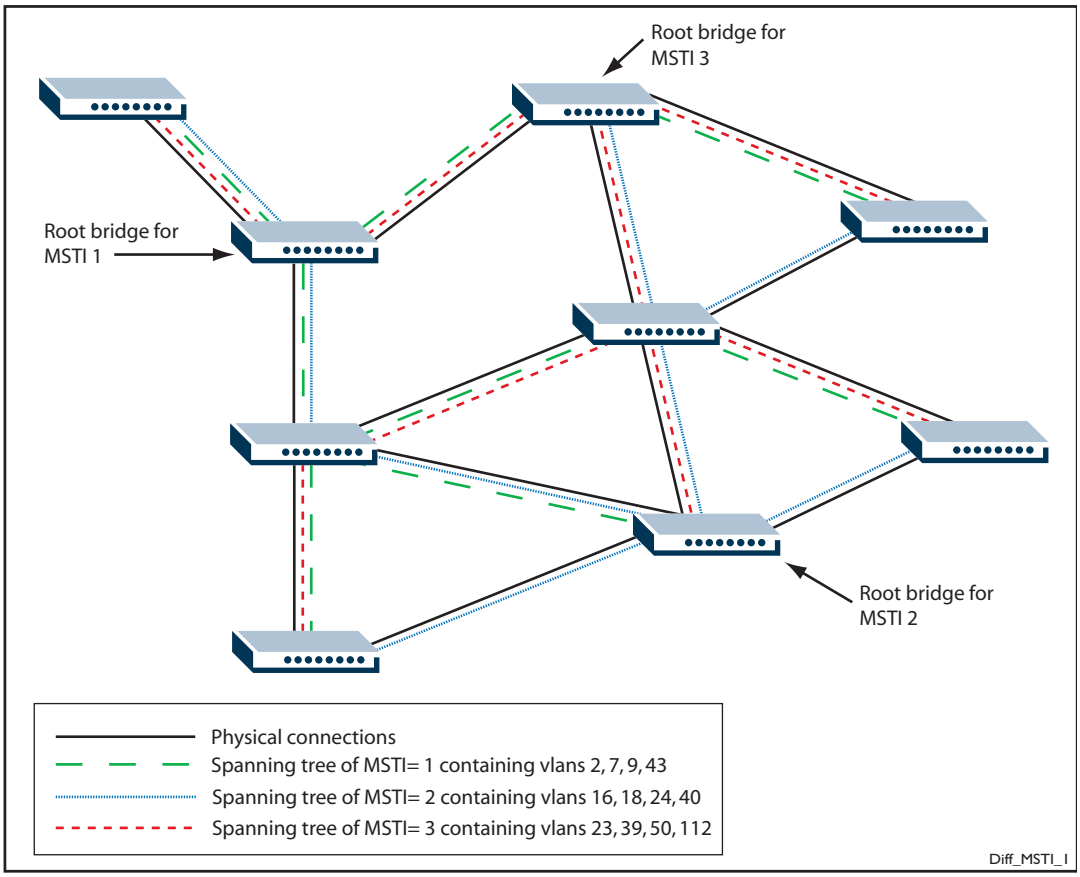
Multiple Spanning Tree Instances (MSTI)

MSTP enables the grouping and mapping of VLANs to different spanning tree instances. So, an MST Instance (MSTI) is a particular set of VLANs that are all using the same spanning tree.

In a network where all VLANs span all links of the network, judicious choice of bridge priorities for different MSTIs can result in different switches becoming root bridges for different MSTIs. That will result in the different MSTIs choosing different active topologies on the network. An example of how different MSTIs can choose different active topologies on the same physical set of links is illustrated in **Figure 20-1**.

MSTP is compatible with RSTP and STP—see **"Common and Internal Spanning Tree (CIST)" on page 20.15**.

Figure 20-1: Different spanning trees created by different MSTIs on the same physical layout



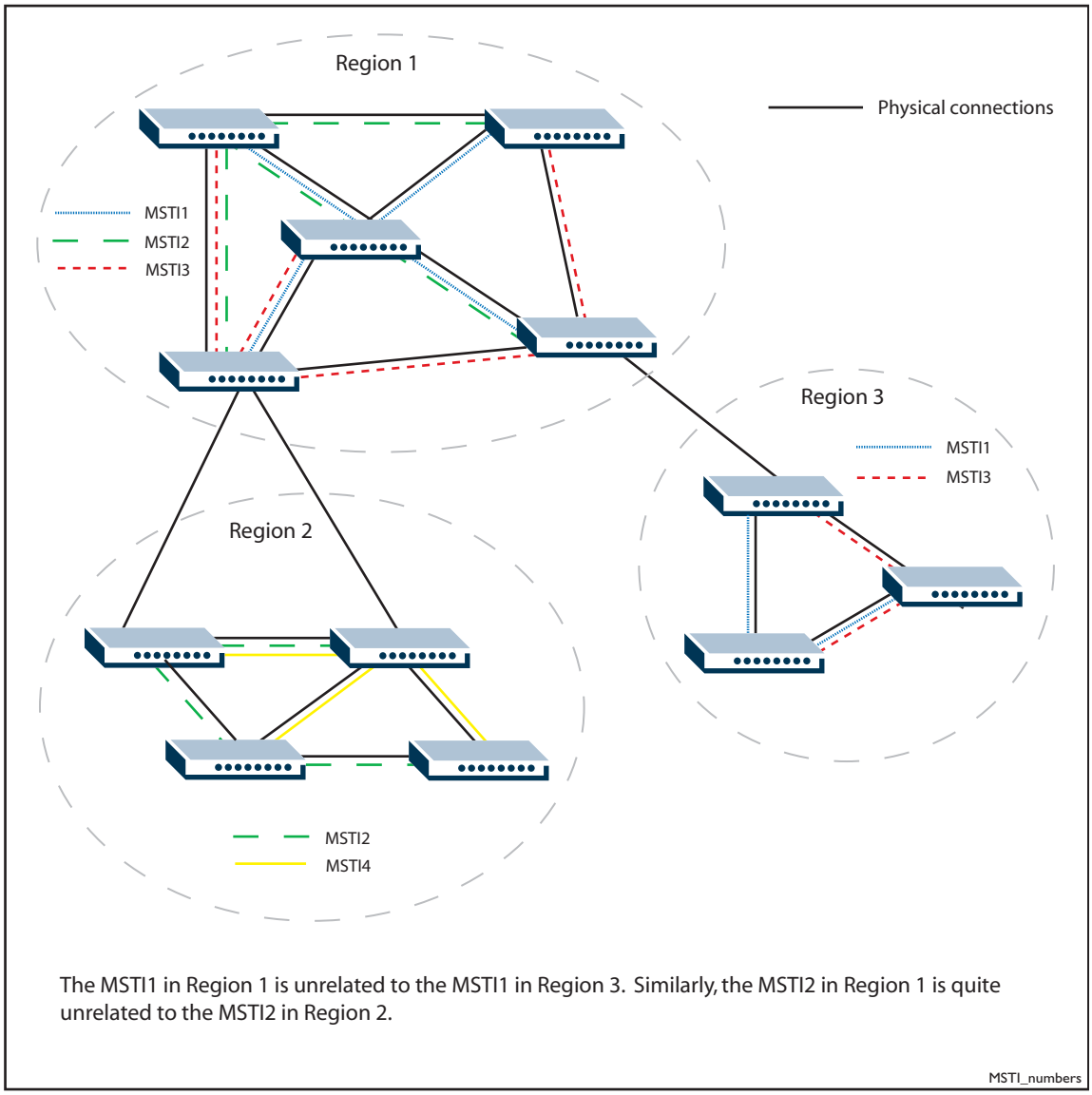
MSTP Regions

An MST region is a set of interconnected switches that all have the same values for the following MST configuration identification elements:

- MST configuration name - the name of the MST region
- Revision level - the revision number of configuration
- Configuration Digest - the mapping of which VLANs are mapped to which MST instances

Each of the MST instances created are identified by an MSTI number. This number is locally significant within the MST region. Therefore, an MSTI will not span across MST regions.

Figure 20-2: MSTIs in different regions



The MSTI1 in Region 1 is unrelated to the MSTI1 in Region 3. Similarly, the MSTI2 in Region 1 is quite unrelated to the MSTI2 in Region 2.

The task of assigning each bridge to a particular region is achieved by the member bridges each comparing their *MST Configuration Identifiers*. More information on configuration identifiers is provided in **Table 20-6**, but for the moment an *MST Configuration Identifier* can simply be thought of as an identifier that represents the mapping of VLANs to MSTIs within each bridge. Therefore, bridges with identical *MST Configuration Identifiers*, must have identical MSTI mapping tables.

While each MSTI can have multiple VLANs, each VLAN can be associated with only one MSTI. Once these associations have been made, the bridges in each region can transmit their spanning tree BPDUs and advertise their MSTIs. This in turn establishes the active data paths between the bridges for each group of VLANs (that is, for each MSTI) and block any duplicate paths within each instance. A particular advantage of this enhancement applies where a large number of VLANs share a few internetwork paths. In this situation there need only be as many Multiple Spanning Tree Instances (MSTIs) as there are source and destination bridge pairs, remembering that a pair of bridges probably has multiple paths between them.

In order to ensure that each bridge within a region maintains the same configuration information (particularly their VID to MSTI mappings) and to ensure each bridge's membership of a particular region, the bridges exchange configuration information in the form of *MST Configuration Identifiers*. **Table 20-6** provides a breakdown of an *MST Configuration Identifier*. A detailed explanation of bridge configuration identifiers can be found in Section 13.7 of the IEEE 802.1Q-2003 standard.

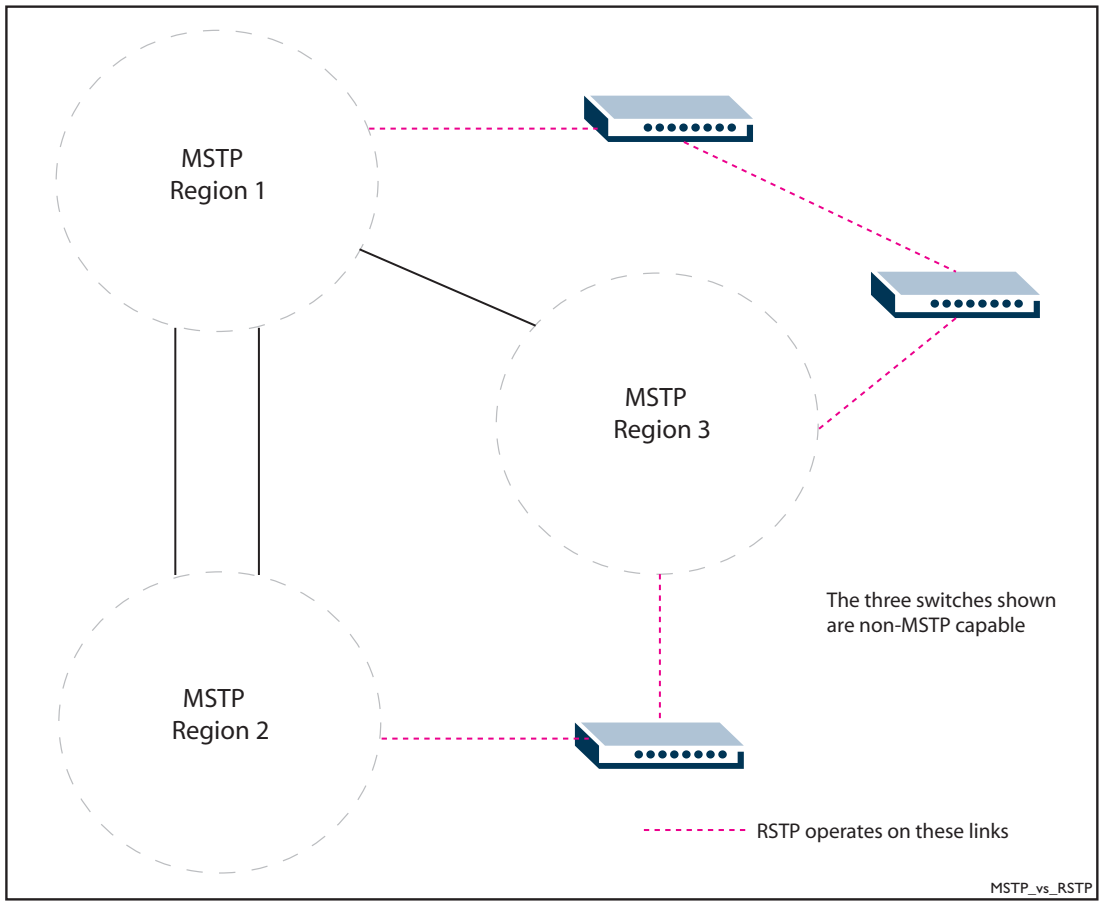
Table 20-6: MST Configuration Identifier

Field Name	Description
Format Selector	A single octet field whose value of 0 indicates MSTP operation
Region Name	A name (up to 32 characters long) that identifies a particular MST region, defined using the **region (MSTP)** command on page 21.10
Revision Level	A number representing the region's revision level, defined using the **revision (MSTP)** command on page 21.11.
Configuration Digest	A 16 octet (HMAC-MD5 based) signature created from the MST configuration table.

Common and Internal Spanning Tree (CIST)

The CIST is the default spanning tree instance of MSTP, i.e. all VLANs that are not members of particular MSTIs are members of the CIST. Also, an individual MST region can be regarded as a single virtual bridge by other MST regions. The spanning tree that runs between regions is the CIST. The CIST is also the spanning tree that runs between MST regions and Single Spanning Tree (SST) entities. So, in **Figure 20-3**, the STP that is running between the regions, and to the SST bridges, is the CIST.

Figure 20-3: The CIST operates on links between regions and to SST devices



Compatibility with Previous Spanning Tree Protocols

MSTP provides for compatibility with older spanning tree protocols in several ways. In addition to the MST region described in the previous section, the protocol provides for single spanning tree systems by employing a Common and Internal Spanning Tree (CIST). The CIST applies a common and internal spanning tree protocol to the whole of the bridged network and is a direct equivalent to the internal spanning tree (IST) protocol of earlier versions.

In common with legacy spanning tree systems, the CIST protocol first determines its root bridge from all the bridges on the network. This is the bridge that contains the lowest bridge identifier. The protocol then selects a regional root bridge for each MSTR. This is the bridge that provides the best path to the CIST root. After the MSTR root bridges have been chosen, they then act on the region's behalf in such a way that the region appears to the Common Spanning Tree (CST) as a virtual bridge. So in addition to having multiple MSTIs, each region operates as a bridge in a CST.

CIST In addition to the individual MSTIs within each MSTP region, the MSTP region is a member of a network-wide spanning tree called the Common and Internal Spanning Tree (CIST). Conceptually, each region represents a virtual bridge. Internal and external bridge connectivity are two independent functions.

Frames with VIDs allocated to the CIST are subject to the rules and path costs of the complete bridged LAN as determined by the CIST's vectors. Frames other than these are subject to the CIST when traveling outside their region, and subject to its particular MSTI inside the region.

The following operational rules apply:

- Each bridge can be a member of only one region.
- A data frame is associated with a single VID.
- Data frames with a given VID are associated with either the CIST or their particular MSTI, but not both.

The role of the Common Spanning Tree (CST) in a network, and the Common and Internal Spanning Tree (CIST) configured on each device, is to prevent loops within a wider network that may span more than one MSTP region and parts of the network running in legacy STP or RSTP mode.

CIST first allocates root and designated bridges by selecting the bridge with the lowest identifier as the root. MSTP then deals with any loops between the regions in the CST. It does this by considering the CIST "vectors" in the following order:

1. CIST External Root Path Cost
2. CIST Regional Root Identifier
3. CIST Internal Root Path Cost
4. CIST Designated Bridge Identifier
5. CIST Designated Port Identifier
6. CIST Receiving Port Identifier

MSTP Bridge Protocol Data Units (BPDUs)

The main function of bridge protocol data units is to enable MSTP to select its root bridges for the CIST (**"Common and Internal Spanning Tree (CIST)" on page 20.15**) and each MSTI. MSTP is compatible with earlier spanning tree versions; its Bridge Protocol Data Unit (BPDU) formats build on earlier versions (**"Compatibility with Previous Spanning Tree Protocols" on page 20.15**).

Table 20-7 shows the standardized format for MSTP BPDU messages. The general format of the BPDUs comprise a common generic portion—octets 1 to 36—that are based on those defined in IEEE Standard 802.1D, 1998, followed by components that are specific to CIST—octets 37 to 102. Components specific to each MSTI are added to this BPDU data block.

Table 20-7: MSTP Bridge Protocol Data Units (BPDUs)

Field Name	Octets	Description
Protocol Identifier	1–2	Protocol being used. The value 0000 0000 0000 0000 identifies the spanning tree algorithm and protocol.
Protocol Version Identifier	3	Identifies the protocol version used.
BPDU Type	4	Value 0000 0000 specifies a configuration BPDU.
CIST Flags	5	Bit 1 is the topology change flag. Bit 2 conveys the CIST proposal flag in RST and MST BPDUs - unused in STP. Bits 3 & 4 convey the CIST port role in RST, and MST BPDUs - unused in STP. Bit 5 conveys the CIST learning flag in RST and MST BPDUs - unused in STP. Bit 6 conveys the CIST forwarding flag in RST and MST BPDUs - unused in STP. Bit 7 conveys the CIST agreement flag in RST and MST BPDUs - unused in STP. Bit 8 conveys the topology change acknowledge flag in STP configuration BPDUs - unused in RSTP and MSTP BPDUs.
CIST Root Identifier	6–13	The Bridge identifier of the CIST Root
CIST External Path Cost	14–17	The path cost between MST regions from the transmitting bridge to the CIST root.
CIST Regional Root Identifier	18–25	ID of the current CIST regional root bridge.
CIST Port Identifier	26–27	CIST port identifier of the transmitting bridge port.
Message Age	28–29	Message age timer value.
Max Age	30–31	Timeout value to be used by all bridges in the bridged network. This value is set by the root. Some implementations of MSTP may choose not to use this value.
Hello Time	32–33	Time interval between the generation of configuration BPDUs by the root bridge.
Forward Delay	34–35	A timeout value used to ensure forward delay timer consistency when transferring a port to the forwarding state. It is also used for ageing filtering database dynamic entries following changes in the active topology.

Table 20-7: MSTP Bridge Protocol Data Units (BPDUs) (cont.)

Field Name	Octets	Description
Version 1 Length	36	Used to convey the Version 1 length. It is always transmitted as 0.
Version 3 Length	37–38	Used to convey the Version 3 length. It is the number of octets taken by the parameters that follow in the BPDU.
MST Configuration Identifier	39–89	An identifier comprising elements of the following: Format Selector Configuration Name Revision Level Configuration Digest.
CIST Internal Root Path Cost	90–93	Path cost to the CIST regional root.
CIST Bridge Identifier	94–101	CIST bridge identifier of the transmitting bridge.
CIST Remaining Hops	102	Remaining hops which limits the propagation and longevity of received spanning tree information for the CIST.
MSTI Configuration Messages (may be absent)	103–39 plus Version 3 Length	See **Table 20-8**.

Table 20-8: MSTI configuration messages

Field Name	Octets	Description
MSTI Flags	1	Bits 1 through 8, convey the topology change flag, proposal flag, port role (two bits), Learning flag, forwarding flag, agreement flag, and master flag for this MSTI.
MSTI Regional Root Identifier	2–9	This includes the value of the MSTID for this configuration message encoded in bits 4 through 1 of octet 1, and bits 8 through 1 of octet 2.
MSTI Internal Root Path Cost	10-13	Internal Root Path Cost.
MSTI Bridge Priority	14	Bits 5 through 8 convey the value of the bridge identifier priority for this MSTI. Bits 1 through 4 of Octet 14 are transmitted as 0, and ignored on receipt.
MSTI Port Priority	15	Bits 5 through 8 are used to convey the value of the port identifier priority for this MSTI. Bits 1 through 4 are transmitted as 0, and ignored on receipt.
MSTI Remaining Hops	16	Value of remaining hops for this MSTI.

Configuring MSTP

By default, RSTP is enabled with default settings on all switch ports. To configure MSTP, see the configuration procedure in **Table 20-9**.

To configure other modes, see **"Configuring RSTP" on page 20.9** or **"Configuring STP" on page 20.6**.

For detailed configuration examples, see the How To Note *How To Configure Basic Switching Functionality*, available from website at **alliedtelesis.com**.

Configuration guidelines for MSTP

- Switches must have the same MST configuration identification elements (region name, revision level and VLAN to MSTI mapping) to be in the same MST region. When configuring multiple MST regions for MSTP, MSTIs are locally significant within an MST region. MSTIs will not span from one region to another region.
- Common and Internal Spanning Tree (CIST) is the default spanning tree instance for MSTP. This means that all VLANs that are not explicitly configured into another MSTI are members of the CIST.
- The software supports a single instance of the MSTP Algorithm consisting of the CIST and up to 15 MSTIs.
- A VLAN can only be mapped to one MSTI or to the CIST. One VLAN mapped to multiple spanning trees is not allowed. All the VLANs are mapped to the CIST by default. Once a VLAN is mapped to a specified MSTI, it is removed from the CIST.
- An MSTI is locally significant within an MST region. An MSTI cannot span across multiple MST regions. The CIST is the spanning tree instance for connecting different MST regions and single spanning tree entities, such as RSTP and STP switches.
- MSTP is compatible with RSTP and STP. An MST region appears as a virtual bridge connecting to single spanning tree entities.
- To avoid unnecessary STP processing, a port that attaches to a LAN that is known to have no other bridges/switches attached can be configured as an edge port.

Before configuring MSTP

Before configuring MSTP, configure VLANs and associate them with switch ports (**Chapter 18, VLAN Introduction** and **Chapter 19, VLAN Commands**), and determine for your network:

- which MSTP regions, revision level and instances are required
- which VLANs and switch ports will belong to which MSTIs,
- which devices you want to be root bridges for each MSTI

Table 20-9: Configuration procedure for MSTP

Command	Description
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `spanning-tree mode mstp`	By default, the device is in RSTP mode. Change to MSTP mode.
`awplus(config)#` `spanning-tree enable`	By default, spanning tree is enabled on all switch ports. If it has been disabled, enable it for MSTP.
Step 1. Configure MSTP region, revision, and instances All MSTP devices in this region of the network must have the same region name, revision number, and VLAN to MSTI mappings.	
`awplus(config)#` `spanning-tree mst configuration`	Enter MST Configuration mode.
`awplus(config-mst)#` `region <region-name>`	Specify the MSTP region. The **region-name** parameter is an arbitrary string that specifies the name you want to assign to the MST region for identification.
`awplus(config-mst)#` `revision <revision-number>`	The **revision-number** parameter specifies the revision of the current MST configuration. The revision is an arbitrary number that you assign to an MST region. It can be used to keep track of the number of times that MST configuration has been updated for the network. Specify the MST revision number in the range 0 to 255.
`awplus(config-mst)#` `instance <msti-id> vlan {<vid>|<vid-list>}`	To allow MSTP to block traffic for different VLANs in different places in a loop, create multiple MSTP instances and associate VLANs with them. Each VLAN can only be in one instance. Specify the MST instance ID in the range 1 to 15.

Table 20-9: Configuration procedure for MSTP (cont.)

Command	Description
Step 2. Advanced configuration The commands above are the minimum required to configure MSTP. The following commands allow more advanced configuration.	
Step 3. Assign root bridge priorities MSTP lets you distribute traffic more efficiently across a network by blocking different links for different VLANs. You do this by making different devices into the root bridge for each MSTP instance, and for the CIST, so that each instance blocks a different link. By default, all devices have the same root bridge priority, 32768 (8000 in hexadecimal), so the device with the lowest MAC address becomes the root bridge. If you want the device to be the root bridge for an instance or for the CIST, set the priority to a lower value (a higher priority) than other devices for this instance. (If you enter a number that is not a multiple of 4096, the device rounds the number down.)	
`awplus(config)#` `spanning-tree mst configuration`	Enter MST Configuration mode.
`awplus(config-mst)#` `instance <msti-id> priority <priority>`	Set the priority for the device to become the root bridge for each instance. Specify the MST instance ID in the range 1 to 15. Specify the root bridge priority in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.
`awplus(config-mst)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `spanning-tree priority <priority>`	Set the priority for the device to become the root bridge for the CIST. Specify the bridge priority in the range 0 to 61440. If you enter a number that is not a multiple of 4096, the switch rounds the number down.
Step 4. Configure edge ports If some switch ports are connected to devices that cannot generate BPDUs (such as workstations), you can set particular switch ports as edge ports, or set them to automatically detect whether they are edge ports.	
`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for these switch ports.

`awplus(config-if)#`
`spanning-tree edgeport (RSTP and MSTP)`
or
`awplus(config-if)#`
`spanning-tree autoedge (RSTP and MSTP)`
	Set these ports to be edge ports,
	or
	set these ports to automatically detect whether they are edge ports.

Table 20-9: Configuration procedure for MSTP (cont.)

Command	Description
Step 5. Configure Root Guard	
`awplus(config-if)#` `spanning-tree guard root`	The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP). Enable the Guard Root feature if required.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
Step 6. Configure BPDU Guard	
`awplus(config)#` `spanning-tree portfast bpdu-guard`	If required, enable the BPDU Guard feature.
`awplus(config)#` `spanning-tree errdisable-timeout enable`	Set a timeout for ports that are disabled due to the BPDU guard feature.
`awplus(config)#` `spanning-tree errdisable-timeout interval <10-1000000>`	Specify the time interval after which a port is brought back up when it has been disabled by the BPDU guard feature.
Step 7. Check MSTP configuration	
`awplus(config)#` `exit`	Return to Privileged Exec mode.
`awplus#` `show spanning-tree mst config`	Check that the digest is the same on this device as for all other devices in the same region.
`awplus#` `show spanning-tree mst`	Check the MST to VLAN and port mapping.
`awplus#` `show spanning-tree mst instance <instance>`	Check the detailed information for a particular instance, and all switch ports associated with that instance. Specify the MST instance ID in the range 1 to 15.
`awplus#` `show spanning-tree mst interface <port>`	Check general information about MSTP, and the CIST settings.

Advanced configuration: For most networks, the default settings of the following will be suitable. However, you can also configure them.

- path costs for ports in an MSTI (**spanning-tree mst instance path-cost**) or for the CIST (**spanning-tree path-cost**)
- port priority for ports in an MSTI (**spanning-tree mst instance priority**) or for the CIST (**spanning-tree priority (port priority)**)

Chapter 21: Spanning Tree Commands

Command List 21.3
clear spanning-tree statistics 21.3
clear spanning-tree detected protocols (RSTP and MSTP) 21.4
debug mstp (RSTP and STP) 21.5
instance priority (MSTP) 21.8
instance vlan (MSTP) 21.9
region (MSTP) 21.10
revision (MSTP) 21.11
show debugging mstp 21.12
show spanning-tree 21.13
show spanning-tree brief 21.16
show spanning-tree mst 21.17
show spanning-tree mst config 21.18
show spanning-tree mst detail 21.19
show spanning-tree mst detail interface 21.20
show spanning-tree mst instance 21.21
show spanning-tree mst instance interface 21.22
show spanning-tree mst interface 21.23
show spanning-tree mst detail interface 21.24
show spanning-tree statistics 21.25
show spanning-tree statistics instance 21.26
show spanning-tree statistics instance interface 21.27
show spanning-tree statistics interface 21.28
show spanning-tree vlan range-index 21.30
spanning-tree autoedge (RSTP and MSTP) 21.31
spanning-tree bpdu 21.32
spanning-tree cisco-interoperability (MSTP) 21.34
spanning-tree edgeport (RSTP and MSTP) 21.35
spanning-tree enable 21.36
spanning-tree errdisable-timeout enable 21.38
spanning-tree errdisable-timeout interval 21.39
spanning-tree force-version 21.40
spanning-tree forward-time 21.41
spanning-tree guard root 21.42
spanning-tree hello-time 21.43
spanning-tree link-type 21.44
spanning-tree max-age 21.45
spanning-tree max-hops (MSTP) 21.46
spanning-tree mode 21.47
spanning-tree mst configuration 21.48
spanning-tree mst instance 21.49
spanning-tree mst instance path-cost 21.50
spanning-tree mst instance priority 21.52
spanning-tree mst instance restricted-role 21.53
spanning-tree mst instance restricted-tcn 21.54
spanning-tree path-cost 21.55
spanning-tree portfast (STP) 21.56

spanning-tree portfast bpdu-filter 21.58
spanning-tree portfast bpdu-guard 21.60
spanning-tree priority (bridge priority) 21.62
spanning-tree priority (port priority) 21.63
spanning-tree restricted-role 21.64
spanning-tree restricted-tcn 21.64
spanning-tree transmit-holdcount 21.65
undebug mstp 21.65

Command List

This chapter provides an alphabetical reference for commands used to configure RSTP, STP or MSTP. For information about spanning trees, including configuration procedures, see **Chapter 20, Spanning Tree Introduction: STP, RSTP, and MSTP**

clear spanning-tree statistics

Use this command to clear all the STP BPDU (Bridge Protocol Data Unit) statistics.

Syntax

```
clear spanning-tree statistics

clear spanning-tree statistics [instance <mstp-instance>]

clear spanning-tree statistics
   [interface <port> [instance <mstp-instance>]]
```

Parameter	Description
<port>	The port to clear STP BPDU statistics for. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).
<mstp-instance>	The MSTP instance (MSTI - Multiple Spanning Tree Instance) to clear MSTP BPDU statistics.

Mode User Exec and Privileged Exec

Usage Use this command with the **instance** parameter in MSTP mode. Specifying this command with the **interface** parameter only not the instance parameter will work in STP and RSTP mode.

Examples

```
awplus# clear spanning-tree statistics

awplus# clear spanning-tree statistics instance 1

awplus# clear spanning-tree statistics interface port1.0.2

awplus# clear spanning-tree statistics interface port1.0.2
       instance 1
```

clear spanning-tree detected protocols (RSTP and MSTP)

Use this command to clear the detected protocols for a specific port, or all ports.

Use this command in RSTP or MSTP mode only.

Syntax `clear spanning-tree detected protocols [interface <port>]`

Parameter	Description
<port>	The port to clear detected protocols for. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode Privileged Exec

Example

```
awplus# clear spanning-tree detected protocols
```

debug mstp (RSTP and STP)

Use this command to enable debugging for the configured spanning tree mode, and echo data to the console, at various levels. Note that although this command uses the keyword **mstp** it displays debugging output for RSTP and STP protocols as well the MSTP protocol.

Use the **no** variant of this command to disable spanning tree debugging.

Syntax

```
debug mstp {all|cli|protocol [detail]|timer [detail]}
debug mstp {packet {rx|tx} [decode] [interface <interface>]}
debug mstp {topology-change [interface <interface>]}
no debug mstp {all|cli|protocol [detail]|timer [detail]}
no debug mstp {packet {rx|tx} [decode] [interface <interface>]}
no debug mstp {topology-change [interface <interface>]}
```

Parameter	Description
`all`	Echoes all spanning tree debugging levels to the console.
`cli`	Echoes spanning tree commands to the console.
`packet`	Echoes spanning tree packets to the console.
`rx`	Received packets.
`tx`	Transmitted packets.
`protocol`	Echoes protocol changes to the console.
`timer`	Echoes timer information to the console.
`detail`	Detailed output.
`decode`	Interprets packet contents
`topology-change`	Interprets topology change messages
`interface`	Keyword before <*interface*> placeholder to specify an interface to debug
`<interface>`	Placeholder used to specify the name of the interface to debug.

Mode Privileged Exec and Global Configuration mode

Usage 1 Use the **debug mstp topology-change interface** command to generate debugging messages when the switch receives an indication of a topology change in a BPDU from another device. The debugging can be activated on a per-port basis. Although this command uses the keyword **mstp**, it displays debugging output for RSTP and STP protocols as well as the MSTP protocol.

Due to the likely volume of output, these debug messages are best viewed using the **terminal monitor** command on page 10.67 before issuing the relevant **debug mstp** command. The default terminal monitor filter will select and display these messages. Alternatively, the messages can be directed to any of the other log outputs by adding a filter for the MSTP application using **log buffered (filter)** command on page 12.9:

```
awplus# configure terminal
awplus(config)# log buffered program mstp
```

Output 1

```
awplus#terminal monitor
awplus#debug mstp topology-change interface port1.0.4
10:09:09 awplus MSTP[1409]: Topology change rcvd on port1.0.4 (internal)
10:09:09 awplus MSTP[1409]: Topology change rcvd on MSTI 1 port1.0.4
aawplus#debug mstp topology-change interface port1.0.6
10:09:29 awplus MSTP[1409]: Topology change rcvd on port1.0.6 (external)
10:09:29 awplus MSTP[1409]: Topology change rcvd on MSTI 1 port1.0.6
```

Usage 2 Use the **debug mstp packet rx|tx decode interface** command to generate debugging messages containing the entire contents of a BPDU displayed in readable text for transmitted and received xSTP BPDUs. The debugging can be activated on a per-port basis and transmit and receive debugging is controlled independently. Although this command uses the keyword **mstp**, it displays debugging output for RSTP and STP protocols as well as the MSTP protocol.

Due to the likely volume of output, these debug messages are best viewed using the **terminal monitor** command on page 10.67 before issuing the relevant **debug mstp** command. The default terminal monitor filter will select and display these messages. Alternatively, the messages can be directed to any of the other log outputs by adding a filter for the MSTP application using the **log buffered (filter)** command on page 12.9:

```
awplus(config)# log buffered program mstp
```

Output 2 In MSTP mode - an MSTP BPDU with 1 MSTI:

```
awplus#terminal monitor
awplus#debug mstp packet rx decode interface port1.0.4
17:23:42 awplus MSTP[1417]: port1.0.4 xSTP BPDU rx - start
17:23:42 awplus MSTP[1417]: Protocol version: MSTP, BPDU type: RST
17:23:42 awplus MSTP[1417]: CIST Flags:  Agree Forward Learn role=Desig
17:23:42 awplus MSTP[1417]: CIST root id      : 0000:0000cd1000fe
17:23:42 awplus MSTP[1417]: CIST ext pathcost : 0
17:23:42 awplus MSTP[1417]: CIST reg root id  : 0000:0000cd1000fe
17:23:42 awplus MSTP[1417]: CIST port id      : 8001 (128:1)
17:23:42 awplus MSTP[1417]: msg age: 0 max age: 20 hellotime: 2 fwd delay: 15
17:23:42 awplus MSTP[1417]: Version 3 length : 80
17:23:42 awplus MSTP[1417]: Format id      : 0
17:23:42 awplus MSTP[1417]: Config name    : test
17:23:42 awplus MSTP[1417]: Revision level : 0
17:23:42 awplus MSTP[1417]: Config digest  : 3ab68794d602fdf43b21c0b37ac3bca8
17:23:42 awplus MSTP[1417]: CIST int pathcost : 0
17:23:42 awplus MSTP[1417]: CIST bridge id    : 0000:0000cd1000fe
17:23:42 awplus MSTP[1417]: CIST hops remaining : 20
17:23:42 awplus MSTP[1417]: MSTI flags           : Agree Forward Learn role=Desig
17:23:42 awplus MSTP[1417]: MSTI reg root id     : 8001:0000cd1000fe
17:23:42 awplus MSTP[1417]: MSTI pathcost        : 0
17:23:42 awplus MSTP[1417]: MSTI bridge priority : 32768 port priority : 128
17:23:42 awplus MSTP[1417]: MSTI hops remaining  : 20
17:23:42 awplus MSTP[1417]: port1.0.4 xSTP BPDU rx - finish
```

In STP mode transmitting a TCN BPDU:

```
awplus#terminal monitor
awplus#debug mstp packet tx decode interface port1.0.4
17:28:09 awplus MSTP[1417]: port1.0.4 xSTP BPDU tx - start
17:28:09 awplus MSTP[1417]: Protocol version: STP, BPDU type: TCN
17:28:09 awplus MSTP[1417]: port1.0.4 xSTP BPDU tx - finish
```

In STP mode receiving an STP BPDU:

```
awplus#terminal monitor
awplus#debug mstp packet rx decode interface port1.0.4
17:31:36 awplus MSTP[1417]: port1.0.4 xSTP BPDU rx - start
17:31:36 awplus MSTP[1417]: Protocol version: STP, BPDU type: Config
17:31:36 awplus MSTP[1417]: Flags:  role=none
17:31:36 awplus MSTP[1417]: Root id      : 8000:0000cd1000fe
17:31:36 awplus MSTP[1417]: Root pathcost : 0
17:31:36 awplus MSTP[1417]: Bridge id  : 8000:0000cd1000fe
17:31:36 awplus MSTP[1417]: Port id      : 8001 (128:1)
17:31:36 awplus MSTP[1417]: msg age: 0 max age: 20 hellotime: 2 fwd delay: 15
17:31:36 awplus MSTP[1417]: ort1.0.4 xSTP BPDU rx - finish
```

In RSTP mode receiving an RSTP BPDU:

```
awplus#terminal monitor
awplus#debug mstp packet rx decode interface port1.0.4
awplus#17:30:17 awplus MSTP[1417]: port1.0.4 xSTP BPDU rx - start
17:30:17 awplus MSTP[1417]: Protocol version: RSTP, BPDU type: RST
17:30:17 awplus MSTP[1417]: CIST Flags:  Forward Learn role=Desig
17:30:17 awplus MSTP[1417]: CIST root id      : 8000:0000cd1000fe
17:30:17 awplus MSTP[1417]: CIST ext pathcost : 0
17:30:17 awplus MSTP[1417]: CIST reg root id  : 8000:0000cd1000fe
17:30:17 awplus MSTP[1417]: CIST port id      : 8001 (128:1)
17:30:17 awplus MSTP[1417]: msg age: 0 max age: 20 hellotime: 2 fwd delay: 15
17:30:17 awplus MSTP[1417]: port1.0.4 xSTP BPDU rx - finish
```

Examples

```
awplus# debug mstp all

awplus# debug mstp cli

awplus# debug mstp packet rx

awplus# debug mstp protocol detail

awplus# debug mstp timer

awplus# debug mstp packet rx decode interface port1.0.2

awplus# debug mstp packet tx decode interface port1.0.6
```

Related Commands

log buffered (filter)
show debugging mstp
terminal monitor
undebug mstp

instance priority (MSTP)

Use this command to set the priority for this device to become the root bridge for the specified MSTI (Multiple Spanning Tree Instance).

Use this command for MSTP only.

Use the **no** variant of this command to restore the root bridge priority of the device for the instance to the default.

Syntax

```
instance <msti-id> priority <priority>
no instance <msti-id> priority
```

Parameter	Description
<msti-id>	Specify the The MST instance ID in the range <1-15>.
<priority>	Specify the root bridge priority for the device for the MSTI in the range <0-61440>. Note that a lower priority number indicates a greater likelihood of the device becoming the root bridge. The priority values can be set only in increments of 4096. If you specify a number that is not a multiple of 4096, it will be rounded down. The default priority is 32768.

Default The default priority value for all instances is 32768.

Mode MST Configuration

Usage MSTP lets you distribute traffic more efficiently across a network by blocking different links for different VLANs. You do this by making different devices into the root bridge for each MSTP instance, so that each instance blocks a different link.
If all devices have the same root bridge priority for the instance, MSTP selects the device with the lowest MAC address to be the root bridge. Give the device a higher priority for becoming the root bridge for a particular instance by assigning it a lower priority number, or vice versa.

Examples To set the root bridge priority for MSTP instance 2 to be the highest (0), so that it will be the root bridge for this instance when available, use the commands:

```
awplus# configure terminal
awplus(config)# spanning-tree mst configuration
awplus(config-mst)# instance 2 priority 0
```

To reset the root bridge priority for instance 2 to the default (32768), use the commands:

```
awplus# configure terminal
awplus(config)# spanning-tree mst configuration
awplus(config-mst)# no instance 2 priority
```

Related Commands
region (MSTP)
revision (MSTP)
show spanning-tree mst config
spanning-tree mst instance
spanning-tree mst instance priority

instance vlan (MSTP)

Use this command to create an MST Instance (MSTI), and associate the specified VLANs with it. An MSTI is a spanning tree instance that exists within an MST region (MSTR). An MSTR can contain up to 15 MSTIs.

When a VLAN is associated with an MSTI the member ports of the VLAN are automatically configured to send and receive spanning-tree information for the associated MSTI. You can disable this automatic configuration of member ports of the VLAN to the associated MSTI by using a **no spanning-tree mst instance** command to remove the member port from the MSTI.

Use the **instance vlan** command for MSTP only.

Use the **no** variant of this command to remove the specified VLANs from the MSTI.

Syntax

```
instance <msti-id> vlan {<vid>|<vid-list>}
no instance <msti-id> vlan {<vid>|<vid-list>}
```

Parameter	Description
<msti-id>	Specify the MST instance ID <1-15>.
<vid>	Specify a VLAN identifier (VID) in the range <1-4094> to be associated with the MSTI specified.
<vid-list>	A hyphen-separated range or a comma-separated list of VLAN IDs

Mode MST Configuration

Usage The VLANs must be created before being associated with an MST instance (MSTI). If the VLAN range is not specified, the MSTI will not be created.

This command removes the specified VLANs from the CIST and adds them to the specified MSTI. If you use the **no** variant of this command to remove the VLAN from the MSTI, it returns it to the CIST. To move a VLAN from one MSTI to another, you must first use the **no** variant of this command to return it to the CIST.

Ports in these VLANs will remain in the control of the CIST until you associate the ports with the MSTI using the **spanning-tree mst instance** command.

Example

```
awplus# configure terminal
awplus(config)# spanning-tree mode mstp
awplus(config)# spanning-tree mst configuration
awplus(config-mst)# instance 2 vlan 30
```

Related Commands

region (MSTP)
revision (MSTP)
show spanning-tree mst config
spanning-tree mst instance
vlan

region (MSTP)

Use this command to assign a name to the device's MST Region. MST Instances (MSTI) of a region form different spanning trees for different VLANs.

Use this command for MSTP only.

Use the **no** variant of this command to remove this region name and reset it to the default.

Syntax

```
region <region-name>
no region
```

Parameter	Description
<region-name>	Specify the name of the region, up to 32 characters. Valid characters are upper-case, lower-case, digits, underscore.

Default By default, the region name is My Name.

Mode MST Configuration

Usage The region name, the revision number, and the digest of the VLAN to MSTI configuration table must be the same on all devices that are intended to be in the same MST region.

Example

```
awplus# configure terminal
awplus(config)# spanning-tree mst configuration
awplus(config-mst)# region ATL
```

Related Commands **revision (MSTP)**
show spanning-tree mst config

revision (MSTP)

Use this command to specify the MST revision number to be used in the configuration identifier.

Use this command for MSTP only.

Syntax `revision <revision-number>`

Parameter	Description
<revision-number>	<0-65535> Revision number.

Default The default of revision number is 0.

Mode MST Configuration

Usage The region name, the revision number, and the digest of the VLAN to MSTI configuration table must be the same on all devices that are intended to be in the same MST region.

Example

```
awplus# configure terminal
awplus(config)# spanning-tree mst configuration
awplus(config-mst)# revision 25
```

Related Commands
region (MSTP)
show spanning-tree mst config
instance vlan (MSTP)

show debugging mstp

Use this command to show the MSTP debugging options set.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging mstp`

Mode User Exec and Privileged Exec mode

Example To display the MSTP debugging options set, enter the command:

```
awplus# show debugging mstp
```

Output **Figure 21-1: Example output from the show debugging mstp command**

```
MSTP debugging status:
  MSTP receiving packet debugging is on
```

Related Commands **debug mstp (RSTP and STP)**

show spanning-tree

Use this command to display detailed spanning tree information on the specified port or on all ports. Use this command for RSTP, MSTP or STP.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree [interface <port-list>]`

Parameter	Description
`interface`	Display information about the following port only.
`<port-list>`	The ports to display information about. A port-list can be: ■ a switch port (e.g. `port1.0.6`) a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix switch ports, static channel groups, and dynamic (LACP) channel groups in the same list

Mode User Exec and Privileged Exec

Usage Note that any list of interfaces specified must not span any interfaces that are not installed.

A topology change counter has been included for RSTP and MSTP. You can see the topology change counter for RSTP by using the **show spanning-tree** command. You can see the topology change counter for MSTP by using the **show spanning-tree mst instance** command.

Example To display spanning tree information about `port1.0.3`, use the command:

```
awplus# show spanning-tree interface port1.0.3
```

Output **Figure 21-2: Example output from the show spanning-tree command**

```
% 1: Bridge up - Spanning Tree Enabled
% 1: Root Path Cost 0 - Root Port 0 -  Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20
% 1: Root Id 80000000cd20f093
% 1: Bridge Id 80000000cd20f093
% 1: last topology change Sun Nov 20 12:24:24 1977
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.3: Port 5023 - Id 839f - Role Designated - State Forwarding
%   port1.0.3: Designated Path Cost 0
%   port1.0.3: Configured Path Cost 200000  - Add type Explicit ref count 1
%   port1.0.3: Designated Port Id 839f - Priority 128  -
%   port1.0.3: Root 80000000cd20f093
%   port1.0.3: Designated Bridge 80000000cd20f093
%   port1.0.3: Message Age 0 - Max Age 20
%   port1.0.3: Hello Time 2 - Forward Delay 15
%   port1.0.3: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 1 - topo change
timer 0
%   port1.0.3: forward-transitions 32
%   port1.0.3: Version Rapid Spanning Tree Protocol - Received None - Send RSTP
%   port1.0.3: No portfast configured - Current  portfast off
%   port1.0.3: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.3: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.3: no root guard configured     - Current root guard off
%   port1.0.3: Configured Link Type point-to-point - Current point-to-point
.
.
```

Figure 21-3: Example output from the show spanning-tree command in RSTP mode

```
awplus#show spanning-tree
% 1: Bridge up - Spanning Tree Enabled
% 1: Root Path Cost 0 - Root Port 0 -  Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20
% 1: Root Id 80000000cd24ff2d
% 1: Bridge Id 80000000cd24ff2d
% 1: last topology change Thu Jul 26 02:06:26 2007
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.1: Port 5001 - Id 8389 - Role Disabled - State Discarding
%   port1.0.1: Designated Path Cost 0
%   port1.0.1: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.1: Designated Port Id 8389 - Priority 128  -
%   port1.0.1: Root 80000000cd24ff2d
%   port1.0.1: Designated Bridge 80000000cd24ff2d
%   port1.0.1: Message Age 0 - Max Age 20
%   port1.0.1: Hello Time 2 - Forward Delay 15
%   port1.0.1: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo change
timer 0
%   port1.0.1: forward-transitions 0
%   port1.0.1: Version Rapid Spanning Tree Protocol - Received None - Send STP
%   port1.0.1: No portfast configured - Current  portfast off
%   port1.0.1: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.1: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.1: no root guard configured     - Current root guard off
%   port1.0.1: Configured Link Type point-to-point - Current  shared
%
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Path Cost 0
%   port1.0.2: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.2: Designated Port Id 838a - Priority 128  -
%   port1.0.2: Root 80000000cd24ff2d
%   port1.0.2: Designated Bridge 80000000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 20
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo change
timer 0
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Rapid Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
```

show spanning-tree brief

Use this command to display a summary of spanning tree status information on all ports. Use this command for RSTP, MSTP or STP.

Syntax `show spanning-tree brief`

Parameter	Description
`brief`	A brief summary of spanning tree information.

Mode User Exec and Privileged Exec

Usage Note that any list of interfaces specified must not span any interfaces that are not installed.

A topology change counter has been included for RSTP and MSTP. You can see the topology change counter for RSTP by using the **show spanning-tree** command. You can see the topology change counter for MSTP by using the **show spanning-tree mst instance** command.

Example To display a summary of spanning tree status information, use the command:

```
awplus# show spanning-tree brief
```

Output **Figure 21-4: Example output from the show spanning-tree brief command**

```
Default: Bridge up - Spanning Tree Enabled
Default: Root Path Cost 40000 - Root Port 4501 -  Bridge Priority 32768
Default: Root Id 8000:0000cd250001
Default: Bridge Id 8000:0000cd296eb1

Port            Designated Bridge    Port Id    Role          State
sa1             8000:001577c9744b    8195       Rootport      Forwarding
po1             8000:0000cd296eb1    81f9       Designated    Forwarding
port1.0.1       8000:0000cd296eb1    8389       Disabled      Discarding
port1.0.2       8000:0000cd296eb1    838a       Disabled      Discarding
port1.0.3       8000:0000cd296eb1    838b       Disabled      Discarding
port1.0.4       8000:0000cd296eb1    838c       Disabled      Discarding
port1.0.5       8000:0000cd296eb1    838d       Disabled      Discarding
port1.0.6       8000:0000cd296eb1    838e       Disabled      Discarding
```

Related Commands **show spanning-tree**

show spanning-tree mst

This command displays bridge-level information about the CIST and VLAN to MSTI mappings.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree mst`

Mode User Exec, Privileged Exec and Interface Configuration

Example To display bridge-level information about the CIST and VLAN to MSTI mappings, enter the command:

```
awplus# show spanning-tree mst
```

Output **Figure 21-5: Example output from the show spanning-tree mst command**

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 - CIST Bridge
Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 8000000475e93ffe
% 1: CIST Reg Root Id 8000000475e93ffe
% 1: CST Bridge Id 8000000475e93ffe
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%
%   Instance        VLAN
%   0:              1
%   2:              4
```

Related Commands **show spanning-tree mst interface**

show spanning-tree mst config

Use this command to display MSTP configuration identifier for the device.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show spanning-tree mst config
```

Mode

User Exec, Privileged Exec and Interface Configuration

Usage

The region name, the revision number, and the digest of the VLAN to MSTI configuration table must be the same on all devices that are intended to be in the same MST region.

Example

To display MSTP configuration identifier information, enter the command:

```
awplus# show spanning-tree mst config
```

Output

Figure 21-6: Example output from the show spanning-tree mst config command

```
awplus#show spanning-tree mst config
%
%  MSTP Configuration Information:
%-----------------------------------------------------------
%  Format Id       : 0
%  Name            : My Name
%  Revision Level  : 0
%  Digest          : 0x80DEE46DA92A98CF21C603291B22880A
%-----------------------------------------------------------
```

Related Commands

instance vlan (MSTP)
region (MSTP)
revision (MSTP)

show spanning-tree mst detail

This command displays detailed information about each instance, and all interfaces associated with that particular instance.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show spanning-tree mst detail`

Mode User Exec, Privileged Exec and Interface Configuration

Example To display detailed information about each instance, and all interfaces associated with them, enter the command:

```
awplus# show spanning-tree mst detail
```

Output **Figure 21-7: Example output from the show spanning-tree mst detail command**

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 -  CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000000cd24ff2d
% 1: CIST Reg Root Id 80000000cd24ff2d
% 1: CIST Bridge Id 80000000cd24ff2d
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.1: Port 5001 - Id 8389 - Role Disabled - State Discarding
%   port1.0.1: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.1: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.1: Designated Port Id 8389 - CIST Priority 128  -
%   port1.0.1: CIST Root 80000000cd24ff2d
%   port1.0.1: Regional Root 80000000cd24ff2d
%   port1.0.1: Designated Bridge 80000000cd24ff2d
%   port1.0.1: Message Age 0 - Max Age 20
%   port1.0.1: CIST Hello Time 2 - Forward Delay 15
%   port1.0.1: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo
change timer 0
.
.
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
%   port1.0.3: Port 5003 - Id 838b - Role Disabled - State Discarding
%   port1.0.3: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.3: Configured Path Cost 20000000  - Add type Explicit ref count 1
%   port1.0.3: Designated Port Id 838b - CIST Priority 128  -
%   port1.0.3: CIST Root 80000000cd24ff2d
%   port1.0.3: Regional Root 80000000cd24ff2d
%   port1.0.3: Designated Bridge 80000000cd24ff2d
%   port1.0.3: Message Age 0 - Max Age 20
%   port1.0.3: CIST Hello Time 2 - Forward Delay 15
%   port1.0.3: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo
change timer 0
%   port1.0.3: forward-transitions 0
%   port1.0.3: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.3: No portfast configured - Current  portfast off
%   port1.0.3: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.3: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.3: no root guard configured     - Current root guard off
%   port1.0.3: Configured Link Type point-to-point - Current  shared
```

show spanning-tree mst detail interface

This command prints detailed information about the specified switch port, and the MST instances associated with it.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show spanning-tree mst detail interface <port>`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode User Exec, Privileged Exec and Interface Configuration

Example To display detailed information about `port1.0.3` and the instances associated with it, enter the command:

```
awplus# show spanning-tree mst detail interface port1.0.3
```

Output **Figure 21-8: Example output from the show spanning-tree mst detail interface command**

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 -  CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000000cd24ff2d
% 1: CIST Reg Root Id 80000000cd24ff2d
% 1: CIST Bridge Id 80000000cd24ff2d
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.2: Configured Path Cost 20000000  - Add type Explicit ref count 2
%   port1.0.2: Designated Port Id 838a - CIST Priority 128  -
%   port1.0.2: CIST Root 80000000cd24ff2d
%   port1.0.2: Regional Root 80000000cd24ff2d
%   port1.0.2: Designated Bridge 80000000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 20
%   port1.0.2: CIST Hello Time 2 - Forward Delay 15
%   port1.0.2: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo
change timer 0
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
% Instance  2:  Vlans: 2
% 1: MSTI Root Path Cost 0 -MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
```

show spanning-tree mst instance

This command displays detailed information for the specified instance, and all switch ports associated with that instance.

A topology change counter has been included for RSTP and MSTP. You can see the topology change counter for RSTP by using the **show spanning-tree** command. You can see the topology change counter for MSTP by using the **show spanning-tree mst instance** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree mst instance <instance>`

Parameter	Description
`<instance>`	Specify an MSTP instance in the range `<1-15>`.

Mode User Exec, Privileged Exec, and Interface Configuration

Usage To display detailed information for **instance 2**, and all switch ports associated with that instance, use the command:

```
awplus# show spanning-tree mst instance 2
```

Output **Figure 21-9: Example output from the show spanning-tree mst instance command**

```
% 1: MSTI Root Path Cost 0 - MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
%
```

show spanning-tree mst instance interface

This command displays detailed information for the specified MST (Multiple Spanning Tree) instance, and the specified switch port associated with that MST instance.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree mst instance <instance> interface <port>`

Parameter	Description
<instance>	Specify an MSTP instance in the range `<1-15>`.
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode User Exec, Privileged Exec, and Interface Configuration

Example To display detailed information for `instance 2`, interface `port1.0.2`, use the command:

```
awplus# show spanning-tree mst instance 2 interface port1.0.2
```

Output **Figure 21-10: Example output from the show spanning-tree mst instance command**

```
% 1: MSTI Root Path Cost 0 - MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
%
```

show spanning-tree mst interface

This command displays the number of instances created, and VLANs associated with it for the specified switch port.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree mst interface <port>`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode User Exec, Privileged Exec, and Interface Configuration

Example To display detailed information about each instance, and all interfaces associated with them, for `port1.0.4`, use the command:

```
awplus# show spanning-tree mst interface port1.0.4
```

Output **Figure 21-11: Example output from the show spanning-tree mst interface command**

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 - CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000008c73a2b22
% 1: CIST Reg Root Id 80000008c73a2b22
% 1: CST Bridge Id 80000008c73a2b22
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 1 sec
%
%   Instance        VLAN
%   0:              1
%   1:              2-3
%   2:              4-5
```

show spanning-tree mst detail interface

This command displays detailed information about the specified switch port, and the MST instances associated with it.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show spanning-tree mst detail interface <port>`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode User Exec, Privileged Exec and Interface Configuration

Example To display detailed information about `port1.0.3` and the instances associated with it, enter the command:

```
awplus# show spanning-tree mst detail interface port1.0.3
```

Output **Figure 21-12: Example output from the show spanning-tree mst detail interface command**

```
% 1: Bridge up - Spanning Tree Enabled
% 1: CIST Root Path Cost 0 - CIST Root Port 0 -  CIST Bridge Priority 32768
% 1: Forward Delay 15 - Hello Time 2 - Max Age 20 - Max-hops 20
% 1: CIST Root Id 80000000cd24ff2d
% 1: CIST Reg Root Id 80000000cd24ff2d
% 1: CIST Bridge Id 80000000cd24ff2d
% 1: portfast bpdu-filter disabled
% 1: portfast bpdu-guard disabled
% 1: portfast errdisable timeout disabled
% 1: portfast errdisable timeout interval 300 sec
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated External Path Cost 0 -Internal Path Cost 0
%   port1.0.2: Configured Path Cost 20000000  - Add type Explicit ref count 2
%   port1.0.2: Designated Port Id 838a - CIST Priority 128  -
%   port1.0.2: CIST Root 80000000cd24ff2d
%   port1.0.2: Regional Root 80000000cd24ff2d
%   port1.0.2: Designated Bridge 80000000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 20
%   port1.0.2: CIST Hello Time 2 - Forward Delay 15
%   port1.0.2: CIST Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0 - topo
change timer 0
%   port1.0.2: forward-transitions 0
%   port1.0.2: Version Multiple Spanning Tree Protocol - Received None - Send STP
%   port1.0.2: No portfast configured - Current  portfast off
%   port1.0.2: portfast bpdu-guard  default  - Current portfast bpdu-guard off
%   port1.0.2: portfast bpdu-filter default  - Current portfast bpdu-filter off
%   port1.0.2: no root guard configured     - Current root guard off
%   port1.0.2: Configured Link Type point-to-point - Current  shared
%
% Instance  2:  Vlans: 2
% 1: MSTI Root Path Cost 0 -MSTI Root Port 0 - MSTI Bridge Priority 32768
% 1: MSTI Root Id 80020000cd24ff2d
% 1: MSTI Bridge Id 80020000cd24ff2d
%   port1.0.2: Port 5002 - Id 838a - Role Disabled - State Discarding
%   port1.0.2: Designated Internal Path Cost 0  - Designated Port Id 838a
%   port1.0.2: Configured Internal Path Cost 20000000
%   port1.0.2: Configured CST External Path cost 20000000
%   port1.0.2: CST Priority 128  - MSTI Priority 128
%   port1.0.2: Designated Root 80020000cd24ff2d
%   port1.0.2: Designated Bridge 80020000cd24ff2d
%   port1.0.2: Message Age 0 - Max Age 0
%   port1.0.2: Hello Time 2 - Forward Delay 15
%   port1.0.2: Forward Timer 0 - Msg Age Timer 0 - Hello Timer 0
```

show spanning-tree statistics

This command displays BPDU (Bridge Protocol Data Unit) statistics for all spanning-tree instances, and all switch ports associated with all spanning-tree instances. For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show spanning-tree statistics`

Mode Privileged Exec

Usage To display BPDU statistics for all spanning-tree instances, and all switch ports associated with all spanning-tree instances, use the command:

```
awplus# show spanning-tree statistics
```

Output **Figure 21-13: Example output from the show spanning-tree statistics command**

```
Port number = 915 Interface = port1.0.6
                ================================
% BPDU Related Parameters
% ------------------------
% Port Spanning Tree                 : Disable
% Spanning Tree Type                 : Rapid Spanning Tree Protocol
% Current Port State                 : Discarding
% Port ID                            : 8393
% Port Number                        : 393
% Path Cost                          : 20000000
% Message Age                        : 0
% Designated Root                    : ec:cd:6d:20:c0:ed
% Designated Cost                    : 0
% Designated Bridge                  : ec:cd:6d:20:c0:ed
% Designated Port Id                 : 8393
% Top Change Ack                     : FALSE
% Config Pending                     : FALSE
% PORT Based Information & Statistics
% ----------------------------------
% Config Bpdu's xmitted              : 0
% Config Bpdu's received             : 0
% TCN Bpdu's xmitted                 : 0
% TCN Bpdu's received                : 0
% Forward Trans Count                : 0
% STATUS of Port Timers
% ---------------------
% Hello Time Configured              : 2
% Hello timer                        : INACTIVE
% Hello Time Value                   : 0
% Forward Delay Timer                : INACTIVE
% Forward Delay Timer Value          : 0
% Message Age Timer                  : INACTIVE
% Message Age Timer Value            : 0
% Topology Change Timer              : INACTIVE
% Topology Change Timer Value        : 0
% Hold Timer                         : INACTIVE
% Hold Timer Value                   : 0
% Other Port-Specific Info
  ------------------------
% Max Age Transitions                : 1
% Msg Age Expiry                     : 0
% Similar BPDUS Rcvd                 : 0
% Src Mac Count                      : 0
% Total Src Mac Rcvd                 : 0
% Next State                         : Learning
% Topology Change Time               : 0
```

show spanning-tree statistics instance

This command displays BPDU (Bridge Protocol Data Unit) statistics for the specified MST (Multiple Spanning Tree) instance, and all switch ports associated with that MST instance. For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree statistics instance <instance>`

Parameter	Description
<instance>	Specify an MSTP instance in the range <1-15>.

Mode Privileged Exec

Usage To display BPDU statistics information for MST instance 2, and all switch ports associated with that MST instance, use the command:

```
awplus# show spanning-tree statistics instance 2
```

Output **Figure 21-14: Example output from the show spanning-tree statistics instance command:**

```
% % INST_PORT port1.0.3 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)    : (0/0)
% Config Bpdu's received (port/inst)   : (0/0)
% TCN Bpdu's xmitted (port/inst)       : (0/0)
% TCN Bpdu's received (port/inst)      : (0/0)
% Message Age(port/Inst)               : (0/0)
% port1.0.3: Forward Transitions                  : 0
% Next State                           : Learning
% Topology Change Time                 : 0
% INST_PORT port1.0.4 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)    : (0/0)
% Config Bpdu's received (port/inst)   : (0/0)
% TCN Bpdu's xmitted (port/inst)       : (0/0)
% TCN Bpdu's received (port/inst)      : (0/0)
% Message Age(port/Inst)               : (0/0)
% port1.0.4: Forward Transitions                  : 0
% Next State                           : Learning
% Topology Change Time                 : 0
% INST_PORT port1.0.5 Information & Statistics
% ----------------------------------------
% Config Bpdu's xmitted (port/inst)    : (0/0)
% Config Bpdu's received (port/inst)   : (0/0)
% TCN Bpdu's xmitted (port/inst)       : (0/0)
% TCN Bpdu's received (port/inst)      : (0/0)
% Message Age(port/Inst)               : (0/0)
% port1.0.5: Forward Transitions                  : 0
% Next State                           : Learning
% Topology Change Time                 : 0%
```

Related Commands **show spanning-tree statistics**

show spanning-tree statistics instance interface

This command displays BPDU (Bridge Protocol Data Unit) statistics for the specified MST (Multiple Spanning Tree) instance and the specified switch port associated with that MST instance.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree statistics instance <instance> interface <port>`

Parameter	Description
<instance>	Specify an MSTP instance in the range `<1-15>`.
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode Privileged Exec

Example To display BPDU statistics for MST `instance 2`, interface `port1.0.2`, use the command:

```
awplus# show spanning-tree statistics instance 2 interface
       port1.0.2
```

Output **Figure 21-15: Example output from the show spanning-tree statistics instance interface command**

```
awplus#sh spanning-tree statistics interface port1.0.2 instance 1
        Spanning Tree Enabled for Instance : 1
        ==================================
% INST_PORT port1.0.2 Information & Statistics
% ---------------------------------------
% Config Bpdu's xmitted (port/inst)    : (0/0)
% Config Bpdu's received (port/inst)   : (0/0)
% TCN Bpdu's xmitted (port/inst)       : (0/0)
% TCN Bpdu's received (port/inst)      : (0/0)
% Message Age(port/Inst)               : (0/0)
% port1.0.2: Forward Transitions                 : 0
% Next State                           : Learning
% Topology Change Time                 : 0

% Other Inst/Vlan Information & Statistics
% --------------------------------------
% Bridge Priority                     : 0
% Bridge Mac Address                  : ec:cd:6d:20:c0:ed
% Topology Change Initiator           : 5023
% Last Topology Change Occured        : Mon Aug 22 05:42:06 2011
% Topology Change                     : FALSE
% Topology Change Detected            : FALSE
% Topology Change Count               : 1
% Topology Change Last Recvd from     : 00:00:00:00:00:00
```

Related Commands **show spanning-tree statistics**

show spanning-tree statistics interface

This command displays BPDU (Bridge Protocol Data Unit) statistics for the specified switch port, and all MST instances associated with that switch port.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree statistics interface <port>`

Parameter	Description
<port>	The port to display information about. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode Privileged Exec

Example To display BPDU statistics about each MST instance for `port1.0.4`, use the command:

```
awplus# show spanning-tree statistics interface port1.0.4
```

Output **Figure 21-16: Example output from the show spanning-tree statistics interface command**

```
awplus#show spanning-tree statistics interface port1.0.2

                Port number = 906 Interface = port1.0.2
                ================================
% BPDU Related Parameters
% ------------------------
% Port Spanning Tree                  : Disable
% Spanning Tree Type                  : Multiple Spanning Tree Protocol
% Current Port State                  : Discarding
% Port ID                             : 838a
% Port Number                         : 38a
% Path Cost                           : 20000000
% Message Age                         : 0
% Designated Root                     : ec:cd:6d:20:c0:ed
% Designated Cost                     : 0
% Designated Bridge                   : ec:cd:6d:20:c0:ed
% Designated Port Id                  : 838a
% Top Change Ack                      : FALSE
% Config Pending                      : FALSE

% PORT Based Information & Statistics
% -----------------------------------
% Config Bpdu's xmitted               : 0
% Config Bpdu's received              : 0
% TCN Bpdu's xmitted                  : 0
% TCN Bpdu's received                 : 0
% Forward Trans Count                 : 0

% STATUS of Port Timers
% ---------------------
% Hello Time Configured               : 2
% Hello timer                         : INACTIVE
% Hello Time Value                    : 0
% Forward Delay Timer                 : INACTIVE
% Forward Delay Timer Value           : 0
% Message Age Timer                   : INACTIVE
% Message Age Timer Value             : 0
% Topology Change Timer               : INACTIVE
% Topology Change Timer Value         : 0
% Hold Timer                          : INACTIVE
% Hold Timer Value                    : 0

% Other Port-Specific Info
  ------------------------
% Max Age Transitions                 : 1
% Msg Age Expiry                      : 0
% Similar BPDUS Rcvd                  : 0
% Src Mac Count                       : 0
% Total Src Mac Rcvd                  : 0
% Next State                          : Learning
% Topology Change Time                : 0


% Other Bridge information & Statistics
  -------------------------------------
% STP Multicast Address               : 01:80:c2:00:00:00
% Bridge Priority                     : 32768
% Bridge Mac Address                  : ec:cd:6d:20:c0:ed
% Bridge Hello Time                   : 2
% Bridge Forward Delay                : 15
% Topology Change Initiator           : 5023
% Last Topology Change Occured        : Mon Aug 22 05:41:20 2011
% Topology Change                     : FALSE
% Topology Change Detected            : TRUE
% Topology Change Count               : 1
% Topology Change Last Recvd from     : 00:00:00:00:00:00
```

Related Commands **show spanning-tree statistics**

show spanning-tree vlan range-index

Use this command to display information about MST (Multiple Spanning Tree) instances and the VLANs associated with them including the VLAN range-index value for the switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show spanning-tree vlan range-index`

Mode Privileged Exec

Example To display information about MST instances and the VLANs associated with them for the switch, including the VLAN range-index value, use the following command:

```
awplus# show spanning-tree vlan range-index
```

Output **Figure 21-17: Example output from the show spanning-tree vlan range-index command**

```
awplus#show spanning-tree vlan range-index
% MST Instance  VLAN       RangeIdx
%      1        1          1
%
```

Related Commands **show spanning-tree statistics**

spanning-tree autoedge (RSTP and MSTP)

Use this command to enable the autoedge feature on the port.

The autoedge feature allows the port to automatically detect that it is an edge port. If it does not receive any BPDUs in the first three seconds after linkup, enabling, or entering RSTP or MSTP mode, it sets itself to be an edgeport and enters the forwarding state.

Use this command for RSTP or MSTP.

Use the **no** variant of this command to disable this feature.

Syntax

```
spanning-tree autoedge
no spanning-tree autoedge
```

Default Disabled

Mode Interface Configuration

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# spanning-tree autoedge
```

Related Commands spanning-tree edgeport (RSTP and MSTP)

spanning-tree bpdu

Use this command in Global Configuration mode to configure BPDU (Bridge Protocol Data Unit) discarding or forwarding, with STP (Spanning Tree Protocol) disabled on the switch.

See the **Usage** note about disabling Spanning Tree before using this command, and using this command to forward unsupported BPDUs unchanged for unsupported STP Protocols.

There is not a **no** variant for this command. Instead, apply the `discard` parameter to reset it back to the default then re-enable STP with **spanning-tree enable** command.

Syntax

```
spanning-tree bpdu {discard|forward|forward-untagged-vlan|
    forward-vlan}
```

Parameter	Description	
`bpdu`	A port that has BPDU filtering enabled will not transmit any BPDUs and will ignore any BPDUs received. This port type has one of the following parameters (in Global Configuration mode):	
	`discard`	Discards all ingress STP BPDU frames.
	`forward`	Forwards any ingress STP BPDU packets to all ports, regardless of any VLAN membership.
	`forward-untagged-vlan`	Forwards any ingress STP BPDU frames to all ports that are untagged members of the ingress port's native VLAN.
	`forward-vlan`	Forwards any ingress STP BPDU frames to all ports that are tagged members of the ingress port's native VLAN.

Default The discard parameter is enabled by default.

Mode Global Configuration

Usage You must first disable Spanning Tree with the **no spanning-tree {mstp|rstp|stp} enable** command before you can use this command to then configure BPDU discarding or forwarding.

This command enables the switch to forward unsupported BPDUs with an unsupported Spanning Tree Protocol, such as proprietary STP protocols with unsupported BPDUs, by forwarding BDPU (Bridge Protocol Data Unit) frames unchanged through the switch.

When you want to revert to default behavior on the switch, issue a **spanning-tree bdpu discard** command and re-enable Spanning Tree with a **spanning-tree enable** command.

Examples To enable STP BPDU discard in Global Configuration mode with STP disabled, which discards all ingress STP BPDU frames, enter the commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree stp enable
awplus(config)# spanning-tree bpdu discard
```

To enable STP BPDU forward in Global Configuration mode with STP disabled, which

forwards any ingress STP BPDU frames to all ports regardless of any VLAN membership, enter the commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree stp enable
awplus(config)# spanning-tree bpdu forward
```

To enable STP BPDU forwarding for untagged frames in Global Configuration mode with STP disabled, which forwards any ingress STP BPDU frames to all ports that are untagged members of the ingress port's native VLAN, enter the commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree stp enable
awplus(config)# spanning-tree bpdu forward-untagged-vlan
```

To enable STP BPDU forwarding for tagged frames in Global Configuration mode with STP disabled, which forwards any ingress STP BPDU frames to all ports that are tagged members of the ingress port's native VLAN, enter the commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree stp enable
awplus(config)# spanning-tree bpdu forward-vlan
```

To reset STP BPDU back to the default `discard` parameter and re-enable STP on the switch, enter the commands:

```
awplus# configure terminal
awplus(config)# spanning-tree bpdu discard
awplus(config)# spanning-tree stp enable
```

Related Commands **show spanning-tree**
spanning-tree enable

spanning-tree cisco-interoperability (MSTP)

Use this command to enable/disable Cisco-interoperability for MSTP.

Use this command for MSTP only.

Syntax `spanning-tree cisco-interoperability {enable|disable}`

Parameter	Description
`enable`	Enable Cisco interoperability for MSTP.
`disable`	Disable Cisco interoperability for MSTP.

Default If this command is not used, Cisco interoperability is disabled.

Mode Global Configuration

Usage For compatibility with certain Cisco devices, all devices in the switched LAN running the AlliedWare Plus™ Operating System must have Cisco-interoperability enabled. When the AlliedWare Plus™ Operating System is interoperating with Cisco, the only criteria used to classify a region are the region name and revision level. VLAN to instance mapping is not used to classify regions when interoperating with Cisco.

Examples To enable Cisco interoperability on a Layer 2 switch:

```
awplus# configure terminal
awplus(config)# spanning-tree cisco-interoperability enable
```

To disable Cisco interoperability on a Layer 2 switch:

```
awplus# configure terminal
awplus(config)# spanning-tree cisco-interoperability disable
```

spanning-tree edgeport (RSTP and MSTP)

Use this command to set a port as an edge-port.

Use this command for RSTP or MSTP.

This command has the same effect as the **spanning-tree portfast (STP)** command, but the configuration displays differently in the output of some show commands.

Use the **no** variant of this command to set a port to its default state (not an edge-port).

Syntax

```
spanning-tree edgeport
no spanning-tree edgeport
```

Default Not an edge port.

Mode Interface Configuration

Usage Use this command on a switch port connected to a LAN that has no other bridges attached. If a BPDU is received on the port that indicates that another bridge is connected to the LAN, then the port is no longer treated as an edge port.

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree edgeport
```

Related Commands **spanning-tree autoedge (RSTP and MSTP)**

spanning-tree enable

Use this command in Global Configuration mode to enable the specified spanning tree protocol for all switch ports. Note that this must be the spanning tree protocol that is configured on the switch by the **spanning-tree mode** command.

Use the **no** variant of this command to disable the configured spanning tree protocol. This places all switch ports in the forwarding state.

Syntax

```
spanning-tree {mstp|rstp|stp} enable
no spanning-tree {mstp|rstp|stp} enable
```

Parameter	Description
`mstp`	Enables or disables MSTP (Multiple Spanning Tree Protocol).
`rstp`	Enables or disables RSTP (Rapid Spanning Tree Protocol).
`stp`	Enables or disables STP (Spanning Tree Protocol).

Default RSTP is enabled by default for all switch ports.

Mode Global Configuration

Usage With no configuration, spanning tree is enabled, and the spanning tree mode is set to RSTP. To change the mode, see **spanning-tree mode** command on page 21.47.

Examples To enable STP in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# spanning-tree stp enable
```

To disable STP in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree stp enable
```

To enable MSTP in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# spanning-tree mstp enable
```

To disable MSTP in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree mstp enable
```

To enable RSTP in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# spanning-tree rstp enable
```

To disable RSTP in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# no spanning-tree rstp enable
```

Related Commands **spanning-tree bpdu**
spanning-tree mode

spanning-tree errdisable-timeout enable

Use this command to enable the errdisable-timeout facility, which sets a timeout for ports that are disabled due to the BPDU guard feature.

Use this command for RSTP or MSTP.

Use the **no** variant of this command to disable the errdisable-timeout facility.

Syntax

```
spanning-tree errdisable-timeout enable
no spanning-tree errdisable-timeout enable
```

Default By default, the errdisable-timeout is disabled.

Mode Global Configuration

Usage The BPDU guard feature shuts down the port on receiving a BPDU on a BPDU-guard enabled port. This command associates a timer with the feature such that the port is re-enabled without manual intervention after a set interval. This interval can be configured by the user using the **spanning-tree errdisable-timeout interval** command.

Example

```
awplus# configure terminal
awplus(config)# spanning-tree errdisable-timeout enable
```

Related Commands

show spanning-tree
spanning-tree errdisable-timeout interval
spanning-tree portfast bpdu-guard

spanning-tree errdisable-timeout interval

Use this command to specify the time interval after which a port is brought back up when it has been disabled by the BPDU guard feature.

Use this command for RSTP or MSTP.

Syntax

```
spanning-tree errdisable-timeout interval <10-1000000>
no spanning-tree errdisable-timeout interval
```

Parameter	Description
<10-1000000>	Specify the errdisable-timeout interval in seconds.

Default By default, the port is re-enabled after 300 seconds.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# spanning-tree errdisable-timeout interval 34
```

Related Commands

show spanning-tree
spanning-tree errdisable-timeout enable
spanning-tree portfast bpdu-guard

spanning-tree force-version

Use this command in Interface Configuration mode for a switch port interface only to force the protocol version for the switch port. Use this command for RSTP or MSTP only.

Syntax

```
spanning-tree force-version <version>
no spanning-tree force-version
```

Parameter	Description	
<version>	<0-3>	Version identifier.
	0	Forces the port to operate in STP mode.
	1	Not supported.
	2	Forces the port to operate in RSTP mode. If it receives STP BPDUs, it can automatically revert to STP mode.
	3	Forces the port to operate in MSTP mode (this option is only available if MSTP mode is configured). If it receives RSTP or STP BPDUs, it can automatically revert to RSTP or STP mode.

Default By default, no version is forced for the port. The port is in the spanning tree mode configured for the device, or a lower version if it automatically detects one.

Mode Interface Configuration mode for a switch port interface only.

Examples Set the value to enforce the spanning tree protocol (STP):

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree force-version 0
```

Set the default protocol version:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree force-version
```

Related Commands show spanning-tree

spanning-tree forward-time

Use this command to set the forward delay value. Use the **no** variant of this command to reset the forward delay value to the default setting of 15 seconds.

The **forward delay** sets the time (in seconds) to control how fast a port changes its spanning tree state when moving towards the forwarding state. If the mode is set to STP, the value determines how long the port stays in each of the listening and learning states which precede the forwarding state. If the mode is set to RSTP or MSTP, this value determines the maximum time taken to transition from discarding to learning and from learning to forwarding.

This value is used only when the switch is acting as the root bridge. Switches not acting as the Root Bridge use a dynamic value for the **forward delay** set by the root bridge. The **forward delay**, **max-age**, and **hello time** parameters are interrelated.

Syntax

```
spanning-tree forward-time <forward-delay>
no spanning-tree forward-time
```

Parameter	Description
<forward-delay	<4-30> The forwarding time delay in seconds.

Default The default is 15 seconds.

Mode Global Configuration

Usage The allowable range for forward-time is 4-30 seconds.

The **forward delay**, **max-age**, and **hello time** parameters should be set according to the following formula, as specified in IEEE Standard 802.1d:

2 x (forward delay - 1.0 seconds) >= max-age

max-age >= 2 x (hello time + 1.0 seconds)

Example

```
awplus# configure terminal
awplus(config)# spanning-tree forward-time 6
```

Related Commands

show spanning-tree
spanning-tree forward-time <forward-delay>
spanning-tree hello-time <hello-time>
spanning-tree mode

spanning-tree guard root

Use this command in Interface Configuration mode for a switch port only to enable the Root Guard feature for the switch port. The root guard feature disables reception of superior BPDUs. You can use this command for RSTP, STP or MSTP.

Use the **no** variant of this command to disable the root guard feature for the port.

Syntax

```
spanning-tree guard root
no spanning-tree guard root
```

Mode Interface Configuration mode for a switch port interface only.

Usage The Root Guard feature makes sure that the port on which it is enabled is a designated port. If the Root Guard enabled port receives a superior BPDU, it goes to a Listening state (for STP) or discarding state (for RSTP and MSTP).

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree guard root
```

spanning-tree hello-time

Use this command to set the hello-time. This sets the time in seconds between the transmission of switch spanning tree configuration information when the switch is the Root Bridge of the spanning tree or is trying to become the Root Bridge.

Use this command for RSTP, STP or MSTP.

Use the **no** variant of this command to restore the default of the hello time.

Syntax

```
spanning-tree hello-time <hello-time>
no spanning-tree hello-time
```

Parameter	Description
<hello-time>	<1-10> The hello BPDU interval in seconds.

Default Default is 2 seconds.

Mode Global Configuration and Interface Configuration for switch ports.

Usage The allowable range of values is 1-10 seconds.

The **forward delay**, **max-age**, and **hello time** parameters should be set according to the following formula, as specified in IEEE Standard 802.1d:

2 x (**forward delay** - 1.0 seconds) >= **max-age**

max-age >= 2 x (**hello time** + 1.0 seconds)

Example

```
awplus# configure terminal
awplus(config)# spanning-tree hello-time 3
```

Related Commands

spanning-tree forward-time <forward-delay>
spanning-tree max-age <max-age>
show spanning-tree

spanning-tree link-type

Use this command in Interface Configuration mode for a switch port interface only to enable or disable point-to-point or shared link types on the switch port.

Use this command for RSTP or MSTP only.

Use the **no** variant of this command to return the port to the default link type.

Syntax

```
spanning-tree link-type {point-to-point|shared}
no spanning-tree link-type
```

Parameter	Description
`shared`	Disable rapid transition.
`point-to-point`	Enable rapid transition.

Default The default link type is point-to-point.

Mode Interface Configuration mode for a switch port interface only.

Usage You may want to set link type to shared if the port is connected to a hub with multiple switches connected to it.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# spanning-tree link-type point-to-point
```

spanning-tree max-age

Use this command to set the max-age. This sets the maximum age, in seconds, that dynamic spanning tree configuration information is stored in the switch before it is discarded.

Use this command for RSTP, STP or MSTP.

Use the **no** variant of this command to restore the default of max-age.

Syntax

```
spanning-tree max-age <max-age>
no spanning-tree max-age
```

Parameter	Description
<max-age>	<6-40> The maximum time, in seconds.

Default The default of spanning-tree max-age is 20 seconds.

Mode Global Configuration

Usage Max-age is the maximum time in seconds for which a message is considered valid.

Configure this value sufficiently high, so that a frame generated by the root bridge can be propagated to the leaf nodes without exceeding the max-age.

The **forward delay**, **max-age**, and **hello time** parameters should be set according to the following formula, as specified in IEEE Standard 802.1d:

2 x (forward delay - 1.0 seconds) >= max-age

max-age >= 2 x (hello time + 1.0 seconds)

Example

```
awplus# configure terminal
awplus(config)# spanning-tree max-age 12
```

Related Commands

show spanning-tree
spanning-tree forward-time <forward-delay>
spanning-tree hello-time <hello-time>

spanning-tree max-hops (MSTP)

Use this command to specify the maximum allowed hops for a BPDU in an MST region. This parameter is used by all the instances of the MST region.

Use the **no** variant of this command to restore the default.

Use this command for MSTP only.

Syntax

```
spanning-tree max-hops <hop-count>
no spanning-tree max-hops <hop-count>
```

Parameter	Description
<hop-count>	Specify the maximum hops the BPDU will be valid for in the range <1-40>.

Default The default max-hops in a MST region is 20.

Mode Global Configuration

Usage Specifying the max hops for a BPDU prevents the messages from looping indefinitely in the network. The hop count is decremented by each receiving port. When a switch receives an MST BPDU that has a hop count of zero, it discards the BPDU.

Examples

```
awplus# configure terminal
awplus(config)# spanning-tree max-hops 25

awplus# configure terminal
awplus(config)# no spanning-tree max-hops
```

spanning-tree mode

Use this command to change the spanning tree protocol mode on the switch. The spanning tree protocol mode on the switch can be configured to either STP, RSTP or MSTP.

Syntax `spanning-tree mode {stp|rstp|mstp}`

Default The default spanning tree protocol mode on the switch is RSTP.

Mode Global Configuration

Usage With no configuration, the switch will have spanning tree enabled, and the spanning tree mode will be set to RSTP. Use this command to change the spanning tree protocol mode on the device. MSTP is VLAN aware, but RSTP and STP are not VLAN aware. To enable or disable spanning tree operation, see the **spanning-tree enable** command on page 21.36.

Examples To change the spanning tree mode from the default of RSTP to MSTP, use the following commands:

```
awplus# configure terminal
awplus(config)# spanning-tree mode mstp
```

Related Commands **spanning-tree enable**

spanning-tree mst configuration

Use this command to enter the MST Configuration mode to configure the Multiple Spanning-Tree Protocol.

Syntax `spanning-tree mst configuration`

Mode Global Configuration

Examples The following example uses this command to enter MST Configuration mode. Note the change in the command prompt.

```
awplus# configure terminal
awplus(config)# spanning-tree mst configuration
awplus(config-mst)#
```

spanning-tree mst instance

Use this command in Interface Configuration mode to assign a Multiple Spanning Tree instance (MSTI) to a switch port or channel group.

Note that ports are automatically configured to send and receive spanning-tree information for the associated MSTI when VLANs are assigned to MSTIs using the **instance vlan (MSTP)** command.

Use the **no** variant of this command in Interface Configuration mode to remove the MSTI from the specified switch port or channel group.

Syntax

```
spanning-tree mst instance <instance-id>
no spanning-tree mst instance <instance-id>
```

Parameter	Description
<instance-id>	<1-15> Specify the MST instance ID. The MST instance must have already been created using the **instance vlan (MSTP)** command.

Default A port automatically becomes a member of an MSTI when it is assigned to a VLAN.

Mode Interface Configuration mode for a switch port or channel group.

Usage You can disable automatic configuration of member ports of a VLAN to an associated MSTI by using a **no spanning-tree mst instance** command to remove the member port from the MSTI. Use the **spanning-tree mst instance** command to add a VLAN member port back to the MSTI.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3

awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3
```

Related Commands

instance vlan (MSTP)
spanning-tree mst instance path-cost
spanning-tree mst instance priority
spanning-tree mst instance restricted-role
spanning-tree mst instance restricted-tcn

spanning-tree mst instance path-cost

Use this command in Interface Configuration mode for a switch port interface only to set the cost of a path associated with a switch port, for the specified MSTI (Multiple Spanning Tree Instance) identifier.

This specifies the switch port's contribution to the cost of a path to the MSTI regional root via that port. This applies when the port is the root port for the MSTI.

Use the **no** variant of this command to restore the default cost value of the path.

Syntax

```
spanning-tree mst instance <instance-id> path-cost <path-cost>
no spanning-tree mst instance <instance-id> path-cost
```

Parameter	Description
<instance-id>	Specify the MSTI identifier in the range <1-15>.
<path-cost>	Specify the cost of path in the range of <1-200000000>, where a lower path-cost indicates a greater likelihood of the specific interface becoming a root.

Default

The default path cost values and the range of recommended path cost values depend on the port speed, as shown in the following table from the IEEE 802.1q-2003 standard.

Port speed	Default path cost	Recommended path cost range
Less than 100 Kb/s	200,000,000	20,000,000-200,000,000
1Mbps	20,000,000	2,000,000-20,000,000
10Mbps	2,000,000	200,000-2,000,000
100 Mbps	200,000	20,000-200,000
1 Gbps	20,000	2,000-20,000
10 Gbps	2,000	200-2, 000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

Mode

Interface Configuration mode for a switch port interface only.

Usage

Before you can use this command to set a path-cost in a VLAN configuration, you must explicitly add an MST instance to a port using the `spanning-tree instance` command.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3 path-cost 1000
```

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3 path-cost
```

Related Commands

instance vlan (MSTP)
spanning-tree mst instance
spanning-tree mst instance priority
spanning-tree mst instance restricted-role
spanning-tree mst instance restricted-tcn

spanning-tree mst instance priority

Use this command in Interface Configuration mode for a switch port interface only to set the port priority for an MST instance (MSTI).

Use the **no** variant of this command to restore the default priority value (128).

Syntax

```
spanning-tree mst instance <instance-id> priority <priority>
no spanning-tree mst instance <instance-id> [priority]
```

Parameter	Description
<instance-id>	Specify the MSTI identifier in the range <1-15>.
<priority>	This must be a multiple of 16 and within the range <0-240>. A lower priority indicates greater likelihood of the port becoming the root port.

Default The default is 128.

Mode Interface Configuration mode for a switch port interface.

Usage This command sets the value of the priority field contained in the port identifier. The MST algorithm uses the port priority when determining the root port for the switch in the MSTI. The port with the lowest value is considered to have the highest priority and will be chosen as root port over a port - equivalent in all other aspects - but with a higher priority value.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3 priority 112

awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3 priority
```

Related Commands

instance vlan (MSTP)
spanning-tree priority (port priority)
spanning-tree mst instance
spanning-tree mst instance path-cost
spanning-tree mst instance restricted-role
spanning-tree mst instance restricted-tcn

spanning-tree mst instance restricted-role

Use this command in Interface Configuration mode for a switch port interface only to enable the restricted role for an MSTI (Multiple Spanning Tree Instance) on a switch port. Configuring the restricted role for an MSTI on a switch port prevents the switch port from becoming the root port in a spanning tree topology.

Use the **no** variant of this command to disable the restricted role for an MSTI on a switch port. Removing the restricted role for an MSTI on a switch port allows the switch port to become the root port in a spanning tree topology.

Syntax

```
spanning-tree mst instance <instance-id> restricted-role
no spanning-tree mst instance <instance-id> restricted-role
```

Parameter	Description
<instance-id>	<1-15> Specify the MST instance ID. The MST instance must have already been created using the **instance vlan (MSTP)** command.

Default The restricted role for an MSTI instance on a switch port is disabled by default.

Mode Interface Configuration mode for a switch port interface only.

Usage The root port is the port providing the best path from the bridge to the root bridge. Use this command to disable a port from becoming a root port. Use the **no** variant of this command to enable a port to become a root port. See **Spanning Tree Operation** for root port information.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3
restricted-role

awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3
restricted-role
```

Related Commands

instance vlan (MSTP)
spanning-tree priority (port priority)
spanning-tree mst instance
spanning-tree mst instance path-cost
spanning-tree mst instance restricted-tcn

spanning-tree mst instance restricted-tcn

Use this command in Interface Configuration mode for a switch port interface only to set the restricted TCN (Topology Change Notification) value to TRUE for the specified MSTI (Multiple Spanning Tree Instance).

Use the **no** variant of this command in Interface Configuration mode to reset the restricted TCN for the specified MSTI to the default value of FALSE.

Syntax

```
spanning-tree mst instance <instance-id> restricted-tcn
no spanning-tree mst instance <instance-id> restricted-tcn
```

Parameter	Description
<instance-id>	<1-15> Specify the MST instance ID. The MST instance must have already been created using the **instance vlan (MSTP)** command.

Default The default value for restricted TCNs is FALSE, as reset with the **no** variant of this command.

Mode Interface Configuration mode for a switch port interface only.

Usage A Topology Change Notification (TCN) is a simple Bridge Protocol Data Unit (BPDU) that a bridge sends out to its root port to signal a topology change. You can configure restricted TCN between TRUE and FALSE values with this command and the **no** variant of this command.

If you configure restricted TCN to TRUE with this command then this stops the switch port from propagating received topology change notifications and topology changes to other switch ports.

If you configure restricted TCN to FALSE with the **no** variant of this command then this enables the switch port to propagate received topology change notifications and topology changes to other switch ports.

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree mst instance 3 restricted-tcn

awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no spanning-tree mst instance 3
                   restricted-tcn
```

Related Commands

instance vlan (MSTP)
spanning-tree priority (port priority)
spanning-tree mst instance
spanning-tree mst instance path-cost
spanning-tree mst instance restricted-role

spanning-tree path-cost

Use this command in Interface Configuration mode for a switch port interface only to set the cost of a path for the specified port. This value then combines with others along the path to the root bridge in order to determine the total cost path value from the particular port, to the root bridge. The lower the numeric value, the higher the priority of the path. This applies when the port is the root port.

Use this command for RSTP, STP or MSTP. When MSTP mode is configured, this will apply to the port's path cost for the CIST.

Syntax

```
spanning-tree path-cost <pathcost>
no spanning-tree path-cost
```

Parameter	Description
<pathcost>	<1-200000000> The cost to be assigned to the port.

Default The default path cost values and the range of recommended path cost values depend on the port speed, as shown in the following table from the IEEE 802.1q-2003 and IEEE 802.1d-2004 standards.

Port speed	Default path cost	Recommended path cost range
Less than 100 Kb/s	200,000,000	20,000,000-200,000,000
1Mbps	20,000,000	2,000,000-20,000,000
10Mbps	2,000,000	200,000-2,000,000
100 Mbps	200,000	20,000-200,000
1 Gbps	20,000	2,000-20,000
10 Gbps	2,000	200-2, 000
100 Gbps	200	20-200
1Tbps	20	2-200
10 Tbps	2	2-20

Mode Interface Configuration mode for switch port interface only.

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree path-cost 123
```

spanning-tree portfast (STP)

Use this command in Interface Configuration mode for a switch port interface only to set a port as an edge-port. The portfast feature enables a port to rapidly move to the forwarding state, without having first to pass through the intermediate spanning tree states. This command has the same effect as the **spanning-tree edgeport (RSTP and MSTP)** command, but the configuration displays differently in the output of some show commands.

Note You can run either of two additional parameters with this command. To simplify the syntax these are documented as separate commands. See the following additional portfast commands:

- **spanning-tree portfast bpdu-filter** command on page 21.58
- **spanning-tree portfast bpdu-guard** command on page 21.60.

You can obtain the same effect by running the **spanning-tree edgeport (RSTP and MSTP)** command. However, the configuration output may display differently in some show commands.

Use the **no** variant of this command to set a port to its default state (not an edge-port).

Syntax

```
spanning-tree portfast

no spanning-tree portfast
```

Default Not an edge port.

Mode Interface Configuration mode for a switch port interface only.

Usage Portfast makes a port move from a blocking state to a forwarding state, bypassing both listening and learning states. The portfast feature is meant to be used for ports connected to end-user devices not switches. Enabling portfast on ports that are connected to a workstation or server allows devices to connect to the network without waiting for spanning-tree to converge.

For example, you may need hosts to receive a DHCP address quickly and waiting for STP to converge would cause the DHCP request to time out. Ensure you do not use portfast on any ports connected to another switch to avoid creating a spanning-tree loop on the network.

Use this command on a switch port that connects to a LAN with no other bridges attached. An edge port should never receive BPDUs. Therefore if an edge port receives a BPDU, the portfast feature takes one of three actions.

- Cease to act as an edge port and pass BPDUs as a member of a spanning tree network (**spanning-tree portfast (STP)** command disabled).
- Filter out the BPDUs and pass only the data and continue to act as a edge port (**spanning-tree portfast bpdu-filter** command enabled).
- Block the port to all BPDUs and data (**spanning-tree portfast bpdu-guard** command enabled).

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree portfast
```

Related Commands

spanning-tree edgeport (RSTP and MSTP)
show spanning-tree
spanning-tree portfast bpdu-filter
spanning-tree portfast bpdu-guard

spanning-tree portfast bpdu-filter

This command sets the bpdu-filter feature and applies a filter to any BPDUs (Bridge Protocol Data Units) received. Enabling this feature ensures that configured ports will not transmit any BPDUs and will ignore (filter out) any BPDUs received. BPDU Filter is not enabled on a port by default.

Using the **no** variant of this command to turn off the bpdu-filter, but retain the port's status as an enabled port. If the port then receives a BPDU it will change its role from an **edge-port** to a **non edge-port**.

Syntax (Global Configuration)

```
spanning-tree portfast bpdu-filter

no spanning-tree portfast bpdu-filter
```

Syntax (Interface Configuration)

```
spanning-tree portfast bpdu-filter {default|disable|enable}

no spanning-tree portfast bpdu-filter
```

Parameter	Description	
`bpdu-filter`	A port that has bpdu-filter enabled will not transmit any BPDUs and will ignore any BPDUs received. This port type has one of the following parameters (in Interface Configuration mode):	
	`default`	Takes the setting that has been configured for the whole switch, i.e. the setting made from the Global configuration mode.
	`disable`	Turns off BPDU filter.
	`enable`	Turns on BPDU filter.

Default BPDU Filter is not enabled on any ports by default.

Mode Global Configuration and Interface Configuration

Usage This command filters the BPDUs and passes only data to continue to act as an edge port. Using this command in Global Configuration mode applies the portfast bpdu-filter feature to all ports on the switch. Using it in Interface mode applies the feature to a specific port, or range of ports.The command will operate in both RSTP and MSTP networks.

Use the **show spanning-tree** command to display status of the bpdu-filter parameter for the switch ports.

Example To enable STP BPDU filtering in Global Configuration mode, enter the commands:

```
awplus# configure terminal
awplus(config)# spanning-tree portfast bpdu-filter
```

To enable STP BPDU filtering in Interface Configuration mode, enter the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree portfast bpdu-filter enable
```

Related Commands
spanning-tree edgeport (RSTP and MSTP)
show spanning-tree
spanning-tree portfast (STP)
spanning-tree portfast bpdu-guard

spanning-tree portfast bpdu-guard

This command applies a BPDU (Bridge Protocol Data Unit) guard to the port. A port with the bpdu-guard feature enabled will block all traffic (BPDUs and user data), if it starts receiving BPDUs.

Use this command in Global Configuration mode to apply BPDU guard to all ports on the switch. Use this command in Interface mode for an individual interface or a range of interfaces specified. BPDU Guard is not enabled on a port by default.

Use the **no** variant of this command to disable the BPDU Guard feature on a switch in Global Configuration mode or to disable the BPDU Guard feature on a port in Interface mode.

Syntax (Global Configuration)

```
spanning-tree portfast bpdu-guard

no spanning-tree portfast bpdu-guard
```

Syntax (Interface Configuration)

```
spanning-tree portfast bpdu-guard {default|disable|enable}

no spanning-tree portfast bpdu-guard
```

Parameter	Description	
`bpdu-guard`	A port that has bpdu-guard turned on will enter the STP blocking state if it receives a BPDU. This port type has one of the following parameters (in Interface Configuration mode):	
	`default`	Takes the setting that has been configured for the whole switch, i.e. the setting made from the Global configuration mode.
	`disable`	Turns off BPDU guard.
	`enable`	Turns on BPDU guard and will also set the port as an edge port.

Default BPDU Guard is not enabled on any ports by default.

Mode Global Configuration or Interface Configuration

Usage This command blocks the port(s) to all BPDUs and data when enabled. BPDU Guard is a port-security feature that changes how a portfast-enabled port behaves if it receives a BPDU. When **bpdu-guard** is set, then the port shuts down if it receives a BPDU. It does not process the BPDU as it is considered suspicious. When **bpdu-guard** is not set, then the port will negotiate spanning-tree with the device sending the BPDUs. By default, bpdu-guard is not enabled on a port.

You can configure a port disabled by the bpdu-guard to re-enable itself after a specific time interval. This interval is set with the **spanning-tree errdisable-timeout interval** command on page 21.39. If you do not use the **errdisable-timeout** feature, then you will need to manually re-enable the port by using the **no shutdown** command.

Use the **show spanning-tree** command on page 21.13 to display the switch and port configurations for the BPDU Guard feature. It shows both the administratively configured and currently running values of bpdu-guard.

Example To enable STP BPDU guard in Global Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# spanning-tree portfast bpdu-guard
```

To enable STP BPDU guard in Interface Configuration mode, enter the below commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree portfast bpdu-guard enable
```

Related Commands
spanning-tree edgeport (RSTP and MSTP)
show spanning-tree
spanning-tree portfast (STP)
spanning-tree portfast bpdu-filter

spanning-tree priority (bridge priority)

Use this command to set the bridge priority for the switch. A lower priority value indicates a greater likelihood of the switch becoming the root bridge.

Use this command for RSTP, STP or MSTP. When MSTP mode is configured, this will apply to the CIST.

Use the **no** variant of this command to reset it to the default.

Syntax

```
spanning-tree priority <priority>
no spanning-tree priority
```

Parameter	Description
<priority>	<0-61440> The bridge priority, which will be rounded to a multiple of 4096.

Default The default priority is 32678.

Mode Global Configuration

Usage To force a particular switch to become the root bridge use a lower value than other switches in the spanning tree.

Example

```
awplus# configure terminal
awplus(config)# spanning-tree priority 4096
```

Related Commands **spanning-tree mst instance priority**
show spanning-tree

spanning-tree priority (port priority)

Use this command in Interface Configuration mode for a switch port interface only to set the port priority for port. A lower priority value indicates a greater likelihood of the port becoming part of the active topology.

Use this command for RSTP, STP, or MSTP. When the device is in MSTP mode, this will apply to the CIST.

Use the **no** variant of this command to reset it to the default.

Syntax

```
spanning-tree priority <priority>
no spanning-tree priority
```

Parameter	Description
<priority>	<0-240>, in increments of 16. The port priority, which will be rounded down to a multiple of 16.

Default The default priority is 128.

Mode Interface Configuration mode for a switch port interface only.

Usage To force a port to be part of the active topology (for instance, become the root port or a designated port) use a lower value than other ports on the device. (This behavior is subject to network topology, and more significant factors, such as bridge ID.)

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree priority 16
```

Related Commands

spanning-tree mst instance priority
spanning-tree priority (bridge priority)
show spanning-tree

spanning-tree restricted-role

Use this command in Interface Configuration mode for a switch port interface only to restrict the port from becoming a root port.

Use the **no** variant of this command to disable the restricted role functionality.

Syntax

```
spanning-tree restricted-role
no spanning-tree restricted-role
```

Default The restricted role is disabled.

Mode Interface Configuration mode for a switch port interface only.

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree restricted-role
```

spanning-tree restricted-tcn

Use this command in Interface Configuration mode for a switch port interface only to prevent TCN (Topology Change Notification) BPDUs (Bridge Protocol Data Units) from being sent on a port. If this command is enabled, after a topology change a bridge is prevented from sending a TCN to its designated bridge.

Use the **no** variant of this command to disable the restricted TCN functionality.

Syntax

```
spanning-tree restricted-tcn
no spanning-tree restricted-tcn
```

Default The restricted TCN is disabled.

Mode Interface Configuration mode for a switch port interface only.

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# spanning-tree restricted-tcn
```

spanning-tree transmit-holdcount

Use this command to set the maximum number of BPDU transmissions that are held back.

Use the **no** variant of this command to restore the default transmit hold-count value.

Syntax

```
spanning-tree transmit-holdcount
no spanning-tree transmit-holdcount
```

Default Transmit hold-count default is 3.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# spanning-tree transmit-holdcount
```

undebug mstp

This command applies the functionality of the **no debug mstp (RSTP and STP)** command.

Chapter 22: Link Aggregation Introduction and Configuration

Introduction and Overview 22.2
Static and Dynamic (LACP) Link Aggregation 22.3
    Static Channel Groups 22.3
    Dynamic (LACP) Channel Groups 22.3
    Link Aggregation Control Protocol (LACP) 22.3
Configuring an LACP Channel Group 22.5
    Minimal LACP Group Configuration 22.8
Configuring a Static Channel Group 22.9

Introduction and Overview

This chapter contains two sample Link Aggregation Control Protocol (LACP), or dynamic channel group, configurations and a sample static channel group configuration.

Link aggregation is the process where two or more ports in an Ethernet switch are combined together to operate as a single virtual port.

Link aggregation is a key component in resilient network design, since it increases the available bandwidth between network devices and it provides continuity of connectivity if one link is broken between network devices.

By aggregating two or more links together, you can increase the bandwidth between neighboring devices since this is effectively additive, where two links give up to twice the bandwidth of one link. Having more than one link to a neighboring device provides connectivity if one of the links break, where a feature of this resiliency is the speed at which link aggregation reacts to the change of link status in a matter of millisecond.

A link aggregation can only exist between a pair of neighboring switches, where the switch ports that are aggregated on one switch cannot be connected to switch ports that are not aggregated on the other switch. A switch can have multiple link aggregations to different neighbors, even to the same neighbor if the network is loop protected.

To see details about the commands used to configure dynamic (LACP) and static Link aggregation, see **Chapter 23, Link Aggregation Commands**.

For a brief overview of static and dynamic link aggregation (LACP), see **Static and Dynamic (LACP) Link Aggregation**.

Static and Dynamic (LACP) Link Aggregation

Channels, either static or dynamic LACP, increase reliability by distributing the data path over more than one physical link. Channels must be configured on both ends of a link or network loops may result. Ports in a channel group need not be contiguous. A mirror port cannot be a member of either a static or a dynamic channel group.

Aggregation criteria

For individual links to be aggregated into a channel group they must:

- originate on the same device or stack
- terminate on the same device or stack
- be members of the same VLANs (**vlan** command on page 19.36)
- have the same data rate (**speed** command on page 17.50)
- share the same admin port key (assigned by using the **channel-group** command on page 23.3 command)
- be operating in full duplex mode (**duplex** command on page 17.11)

The hardware must also be capable and have the capacity to handle the number of links to be aggregated.

Static Channel Groups

A static channel group, also known as a static aggregator, enables a number of ports to be manually configured to form a single logical connection of higher bandwidth. By using static channel groups you increase channel reliability by distributing the data path over more than one physical link. Static channel groups are best used in simpler environments, usually where neighbor switches are close together, situated within the same rack, so that you can easily ensure that the correct statically aggregated ports are connected together.

For a static channel group configuration example see the **Configuring a Static Channel Group** section in this chapter. For details of static channel group commands, such as the **static-channel-group** command, see **Chapter 23, Link Aggregation Commands**.

Dynamic (LACP) Channel Groups

A LACP channel group, also known as an etherchannel, a LACP aggregator, or a dynamic channel group, enables a number of ports to be dynamically combined to form a single higher bandwidth logical connection. LACP channel groups are best used for complex environments, typically long-distance links, to detect failure between neighbor switches.

For LACP configuration examples see **Configuring an LACP Channel Group** and **Minimal LACP Group Configuration** sections in this chapter. For details of LACP channel group commands, such as the **channel-group** command, see **Chapter 23, Link Aggregation Commands**.

Link Aggregation Control Protocol (LACP)

LACP is based on the IEEE Standard 802.3ad. It allows bundling of several physical ports to form a single logical channel providing enhanced performance and resiliency. The aggregated channel is viewed as a single link by each switch. Spanning tree also views the channel as one interface and not as multiple interfaces. When there is a failure in one physical port, the other ports stay up and there is no disruption.

Note

AlliedWare Plus™ supports IEEE 802.3ad link aggregation and uses the Link Aggregation Control Protocol (LACP). LACP does not interoperate with devices that use Port Aggregation Protocol (PAgP).

Note

Link aggregation does not necessarily achieve exact load balancing across the links. The load sharing algorithm is designed to ensure that any given data flow always goes down the same link. It also aims to spread data flows across the links as evenly as possible.

Link aggregation hashes one or more of the source and destination MAC address, IP address and UDP/TCP ports to select a link on which to send a packet. So packet flow between a pair of hosts always takes the same link inside the Link Aggregation Group (LAG). The net effect is that the bandwidth for a given packet stream is restricted to the speed of one link in the LAG.

For example, for a 2 Gbps LAG that is a combination of two 1 Gbps ports, any one flow of traffic can only ever reach a maximum throughput of 1 Gbps. However, the hashing algorithm should spread the flows across the links so that when many flows are operating, the full 2 Gbps can be utilized.

For information about load balancing see the **platform load-balancing** command.

LACP operates where systems are connected over multiple communications links. Once LACP has been initially configured and enabled, it automatically aggregates the ports that have been assigned to a channel group, if possible. LACP continues to monitor these groups and dynamically adds or removes links to them as network changes occur.

LACP achieves this by determining:

- which ports are under LACP control (**channel-group** command on page 23.3)
- whether each port is in LACP active or LACP passive mode (**channel-group** command on page 23.3)
- which system has the highest LACP priority (**lacp system-priority** command on page 23.8)
- the LACP priority of ports (**lacp port-priority** command on page 23.7)
- whether the LACP timeout is short or long (**lacp timeout** command on page 23.9)

Channel group identification

In order to identify particular channel groups, each group is assigned a link aggregation identifier called a **lag ID**. The lag ID comprises the following components for both the local system (called the Actor) followed by their equivalent components for the remote system (called the Partner):

- system identifier - the MAC address of the system
- port key - An identifier - created by the LACP software
- port priority - set by the **lacp port-priority** command on page 23.7
- port number - determined by the device connection

The lag ID can be displayed for each aggregated link by entering the **show etherchannel** command on page 23.13.

Configuring an LACP Channel Group

The following example shows how to configure three links between two Allied Telesis managed Layer 3 Switches. The three links are assigned the same administrative key (1), so that they aggregate to form a single channel (1). They are viewed by the STP as one interface.

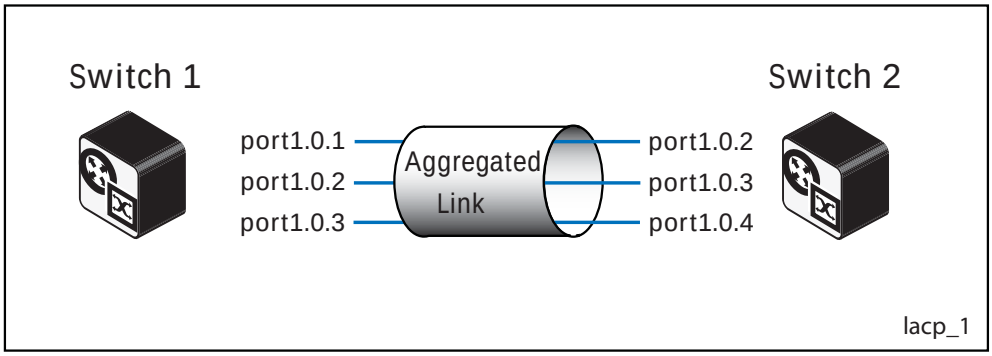


Table 22-1: Switch 1 configuration

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `lacp system-priority 20000`	Set the system priority of this switch. This priority is used to determine which switch in the system is responsible for resolving conflicts in the choice of aggregation groups. A lower numerical value has a higher priority. Switch 1 has a higher priority than Switch 2 in this configuration.
`awplus(config)#` `interface port1.0.1`	Enter the Interface Configuration mode to configure port `1.0.1`.
`awplus(config-if)#` `channel-group 1 mode active`	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configure mode.
`awplus(config)#` `interface port1.0.2`	Enter the Interface Configuration mode to configure port `1.0.2`.
`awplus(config-if)#` `channel-group 1 mode active`	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.

Table 22-1: Switch 1 configuration (cont.)

`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configure mode.
`awplus(config)#` `interface port1.0.3`	Enter the Interface Configuration mode to configure port `1.0.3`.
`awplus(config-if)#` `channel-group 1 mode active`	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.
`awplus(config-if)#` `interface po1`	Select the dynamic aggregator logical interface created for channel-group 1 named `po1`.

Table 22-2: Switch 2 configuration

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `lacp system-priority 3000`	Set the system priority of this switch. This priority is used to determine which switch in the system is responsible for resolving conflicts in the choice of aggregation groups. A lower numerical value has a higher priority. Switch 2 has a lower priority than Switch 1 in this configuration.
`awplus(config)#` `interface port1.0.2`	Enter the Interface Configuration mode to configure port `1.0.2`.
`awplus(config-if)#` `channel-group 1 mode active`	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.
`awplus(config-if)#` `exit`	Exit the Interface mode and return to the Configure mode.
`awplus(config)#` `interface port1.0.3`	Enter the Interface mode to configure port `1.0.3`.
`awplus(config-if)#` `channel-group 1 mode active`	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.

Table 22-2: Switch 2 configuration

`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `interface port1.0.4`	Enter the Interface Configuration mode to configure port `1.0.4`.
`awplus(config-if)#` `channel-group 1 mode active`	Add this interface to a channel group `1` and enable link aggregation so that it may be selected for aggregation by the local system.
`awplus(config-if)#` `interface po1`	Select the dynamic aggregator logical interface created for channel-group 1 named `po1`.

Commands used

lacp system-priority
channel-group

Validation commands

show lacp sys-id
show port etherchannel
show etherchannel
show etherchannel detail

Minimal LACP Group Configuration

For details of LACP channel group commands, see **Chapter 23, Link Aggregation Commands**.

The following minimal LACP group configuration example creates LACP channel group 2 and enables link aggregation on switch ports `1.0.1` and `1.0.2` within this channel group. Note that all aggregated ports must belong to the same VLAN.

`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `interface port1.0.1-port1.0.2`	Enter the Interface Configuration mode for the switch ports to aggregate into the channel group.
`awplus(config-if)#` `channel-group 2 mode active`	Assign the switch ports to channel group `2` in active mode. This creates the channel group.
`awplus(config-if)#` `interface po2`	Select the dynamic aggregator logical interface created for channel-group 2 named `po2`.

Commands used

channel-group

Validation commands

show static-channel-group

Configuring a Static Channel Group

For details of LACP channel group commands, see **Chapter 23, Link Aggregation Commands**.

The following example creates a static channel group and adds switch ports `1.0.1` and `1.0.2`.

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `interface port1.0.1`	Enter the Interface Configuration mode to configure port `1.0.1`.
`awplus(config-if)#` `static-channel-group 2`	Add port `1.0.1` to static-channel-group `2`.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `interface port1.0.2`	Enter the Interface Configuration mode to configure port `1.0.2`.
`awplus(config-if)#` `static-channel-group 2`	Add port `1.0.2` to static-channel-group `2`.
`awplus(config-if)#` `interface sa2`	Select the static aggregator logical interface created for static-channel-group 2 named `sa2`.

Commands used

static-channel-group

Validation commands

show static-channel-group

Chapter 23: Link Aggregation Commands

Introduction 23.2
Command List 23.3
channel-group 23.3
clear lacp counters 23.5
debug lacp 23.6
lacp port-priority 23.7
lacp system-priority 23.8
lacp timeout 23.9
show debugging lacp 23.11
show diagnostic channel-group 23.12
show etherchannel 23.13
show etherchannel detail 23.14
show etherchannel summary 23.15
show lacp sys-id 23.16
show lacp-counter 23.17
show port etherchannel 23.18
show static-channel-group 23.19
static-channel-group 23.20
undebug lacp 23.21

Introduction

This chapter provides an alphabetical reference of commands used to configure a static channel group (static aggregator) and dynamic channel group (LACP channel group, etherchannel or LACP aggregator). Link aggregation is also sometimes referred to as channeling.

Note

AlliedWare Plus™ supports IEEE 802.3ad link aggregation and uses the Link Aggregation Control Protocol (LACP). LACP does not interoperate with devices that use Port Aggregation Protocol (PAgP).

Note

Link aggregation does not necessarily achieve exact load balancing across the links. The load sharing algorithm is designed to ensure that any given data flow always goes down the same link. It also aims to spread data flows across the links as evenly as possible.

Link aggregation hashes one or more of the source and destination MAC address, IP address and UDP/TCP ports to select a link on which to send a packet. So packet flow between a pair of hosts always takes the same link inside the Link Aggregation Group (LAG). The net effect is that the bandwidth for a given packet stream is restricted to the speed of one link in the LAG.

For example, for a 2 Gbps LAG that is a combination of two 1 Gbps ports, any one flow of traffic can only ever reach a maximum throughput of 1 Gbps. However, the hashing algorithm should spread the flows across the links so that when many flows are operating, the full 2 Gbps can be utilized.

For information about load balancing see the **platform load-balancing** command.

For a description of static and dynamic link aggregation (LACP), see **"Configuring an LACP Channel Group" on page 22.5**. For an LACP configuration example, see **Chapter 22, Link Aggregation Introduction and Configuration**.

Command List

channel-group

Use this command to add the switch port to a dynamic channel group specified by the dynamic channel group number, and set its mode. This command enables LACP link aggregation on the switch port, so that it may be selected for aggregation by the local system. Dynamic channel groups are also known as LACP channel groups, LACP aggregators or etherchannels.

You can create up to 32 dynamic (LACP) channel groups (and up to 96 static channel groups).

Use the **no** variant of this command to turn off link aggregation on the switch port. You will be returned to Global Configuration mode from Interface Configuration mode.

Syntax

```
channel-group <dynamic-channel-group-number> mode {active|passive}
no channel-group
```

Parameter	Description
<dynamic-channel-group-number>	<1-32> Specify a dynamic channel group number for an LACP link. You can create up to 32 dynamic (LACP) channel groups (as well as up to 96 static channel groups).
active	Enables initiation of LACP negotiation on a port. The port will transmit LACP dialogue messages whether or not it receives them from the partner system.
passive	Disables initiation of LACP negotiation on a port. The port will only transmit LACP dialogue messages if the partner systems is transmitting them, i.e., the partner is in the active mode.

Mode Interface Configuration

Usage All the switch ports in a channel-group must belong to the same VLANs, have the same tagging status, and can only be operated on as a group. All switch ports within a channel group must have the same port speed and be in full duplex mode.

Once the LACP channel group has been created, it is treated as a switch port, and can be referred to in most other commands that apply to switch ports.

To refer to an LACP channel group in other LACP commands, use the channel group number. To specify an LACP channel group (LACP aggregator) in other commands, prefix the channel group number with **po**. For example, 'po2' refers to the LACP channel group with channel group number 2.

For more on LACP, see **"Dynamic (LACP) Channel Groups" on page 22.3** and **Chapter 22, Link Aggregation Introduction and Configuration**.

Examples

To add switch `port1.0.6` to a newly created LACP channel group 2 use the commands below:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# channel-group 2 mode active
```

To remove switch `port1.0.6` from any created LACP channel groups use the command below:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# no channel-group
awplus(config)#
```

To reference the pre-defined LACP channel group 2 as an interface, apply commands as below:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# channel-group 2 mode active
awplus(config-if)# exit
awplus(config)# interface port.1.0.6
awplus(config-if)# channel-group 2 mode active
awplus(config-if)# exit
awplus(config)# interface po2
awplus(config-if)#
```

Related Commands

show etherchannel
show etherchannel detail
show etherchannel summary
show port etherchannel

clear lacp counters

Use this command to clear all counters of all present LACP aggregators (channel groups) or a given LACP aggregator.

Syntax `clear lacp [<1-32>] counters`

Parameter	Description
<1-32>	Channel-group number.

Mode Privileged Exec

Example

```
awplus# clear lacp 2 counters
```

debug lacp

Use this command to enable all LACP troubleshooting functions.

Use the **no** variant of this command to disable this function.

Syntax

```
debug lacp {all|cli|event|ha|packet|sync|timer[detail]}
no debug lacp {all|cli|event|ha|packet|sync|timer[detail]}
```

Parameter	Description
`all`	Turn on all debugging for LACP.
`cli`	Specifies debugging for CLI messages. Echoes commands to the console.
`event`	Specifies debugging for LACP events. Echoes events to the console.
`ha`	Specifies debugging for HA (High Availability) events. Echoes High Availability events to the console.
`packet`	Specifies debugging for LACP packets. Echoes packet contents to the console.
`sync`	Specified debugging for LACP synchronization. Echoes synchronization to the console.
`timer`	Specifies debugging for LACP timer. Echoes timer expiry to the console.
`detail`	Optional parameter for LACP timer-detail. Echoes timer start/stop details to the console.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# debug lacp timer detail
awplus# debug lacp all
```

Related Commands **show debugging lacp**
undebug lacp

lacp port-priority

Use this command to set the priority of a switch port. Ports are selected for aggregation based on their priority, with the higher priority (numerically lower) ports selected first.

Use the **no** variant of this command to reset the priority of port to the default.

Syntax

```
lacp port-priority <1-65535>
no lacp port-priority
```

Parameter	Description
<1-65535>	Specify the LACP port priority.

Default The default is 32768.

Mode Interface Configuration

Example

```
awplus# configure terminal
awplus(config)# interface port1.0.5
awplus(config-if)# lacp port-priority 34
```

lacp system-priority

Use this command to set the system priority of a local system. This is used in determining the system responsible for resolving conflicts in the choice of aggregation groups.

Use the **no** variant of this command to reset the system priority of the local system to the default.

Syntax

```
lacp system-priority <1-65535>
no lacp system-priority
```

Parameter	Description
<1-65535>	LACP system priority. Lower numerical values have higher priorities.

Default The default is 32768.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# lacp system-priority 6700
```

lacp timeout

Use this command to set the short or long timeout on a port. Ports will time out of the aggregation if three consecutive updates are lost.

Syntax `lacp timeout {short|long}`

Parameter	Description
`timeout`	Number of seconds before invalidating a received LACP data unit (DU).
`short`	LACP short timeout. The **short** timeout value is **1** second.
`long`	LACP long timeout. The **long** timeout value is **30** seconds.

Default The default is **long** timeout (30 seconds).

Mode Interface Configuration

Usage This command enables the switch to indicate the rate at which it expects to receive LACPDUs from its neighbor.

If the timeout is set to **long**, then the switch expects to receive an update every **30** seconds, and this will time a port out of the aggregation if no updates are seen for 90 seconds (i.e. 3 consecutive updates are lost).

If the timeout is set to **short**, then the switch expects to receive an update every second, and this will time a port a port out of the aggregation if no updates are seen for 3 seconds (i.e. 3 consecutive updates are lost).

The switch indicates its preference by means of the '`Timeout`' field in the '`Actor`' section of its LACPDUs. If the '`Timeout`' field is set to 1, then the switch has set the **short** timeout. If the '`Timeout`' field is set to 0, then the switch has set the **long** timeout.

Setting the **short** timeout enables the switch to be more responsive to communication failure on a link, and does not add too much processing overhead to the switch (1 packet per second).

Note It is not possible to configure the rate that the switch sends LACPDUs; the switch must send at the rate which the neighbor indicates it expects to receive LACPDUs.

Examples The following commands set the LACP long timeout period for 30 seconds on `port1.0.2`.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# lacp timeout long
```

The following commands set the LACP short timeout for 1 second on `port1.0.2`.

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# lacp timeout short
```

show debugging lacp

Use this command to display the LACP debugging option set.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show debugging lacp`

Mode User Exec and Privileged Exec

Example

```
awplus# show debugging lacp
```

Output **Figure 23-1: Example output from the show debugging lacp command**

```
LACP debugging status:
 LACP timer debugging is on
 LACP timer-detail debugging is on
 LACP cli debugging is on
 LACP packet debugging is on
 LACP event debugging is on
 LACP sync debugging is on
```

Related Commands debug lacp

show diagnostic channel-group

This command displays dynamic and static channel group interface status information. The output of this command is useful for Allied Telesis authorized service personnel for diagnostic purposes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show diagnostic channel-group`

Mode User Exec and Privileged Exec

Example

```
awplus# show diagnostic channel-group
```

Output **Figure 23-2: Example output from the show diagnostic channel-group command**

```
awplus#show diagnostic channel-group

Channel Group Info based on NSM:
Note: Pos - position in hardware table
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503     port1.0.15   5015     No
     sa3        4503     port1.0.18   5018     No
     po1        4601     port1.0.7    5007     No
     po1        4601     port1.0.8    5008     No
     po1        4601     port1.0.9    5009     No


Channel Group Info based on HSL:
Note: Pos - position in hardware table
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503                           N/a
     po1        4601                           N/a


Channel Group Info based on IPIFWD:
Note: Pos - position in hardware table
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503                           N/a
     po1        4601                           N/a


Channel Group Info based on HW:
Note: Pos - position in hardware table
      Only entries from first device are displayed.
-------------------------------------------------------------
Dev  Interface  IfIndex  Member port  IfIndex  Active  Pos
-------------------------------------------------------------
     sa3        4503                           N/a
     po1        4601                           N/a


No error found
```

Related Commands **show tech-support**

show etherchannel

Use this command to display information about a LACP channel specified by the channel group number.

The command output also shows the thrash limiting status. If thrash limiting is detected and the **thrash limiting** parameter of the **thrash-limiting** command on page 17.57 is set to **vlan disable**, the output will also show the VLANs on which thrashing is detected.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show etherchannel [<1-32>]`

Parameter	Description
<1-32>	Channel-group number.

Mode User Exec and Privileged Exec

Example

```
awplus# show etherchannel 2
```

Output **Figure 23-3: Example output from the show etherchannel command for a particular channel**

```
% Lacp Aggregator: po1
 Thrash-limiting
 Status Vlan Thrashing Detected, Action vlan-disable 60(s)
 Thrashing Vlans 1 2 3 4 5
% Member:
   port1.0.4
   port1.0.6
```

show etherchannel detail

Use this command to display detailed information about all LACP channels.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show etherchannel detail`

Mode User Exec and Privileged Exec

Example

```
awplus# show etherchannel detail
```

Output **Figure 23-4: Example output from the show etherchannel detail command**

```
% Aggregator po1 (4501)
%  Mac address: 00:00:cd:24:fd:29
%  Admin Key: 0001 - Oper Key 0001
%  Receive link count: 1 - Transmit link count: 0
%  Individual: 0 - Ready: 1
%  Partner LAG: 0x8000,00-00-cd-24-da-a7
%   Link: port1.0.1 (5001) disabled
%   Link: port1.0.2 (5002) sync: 1
% Aggregator po2 (4502)
%  Mac address: 00:00:cd:24:fd:29
%  Admin Key: 0002 - Oper Key 0002
%  Receive link count: 1 - Transmit link count: 0
%  Individual: 0 - Ready: 1
%  Partner LAG: 0x8000,00-00-cd-24-da-a7
%   Link: port1.0.6 (5007) disabled
```

show etherchannel summary

Use this command to display a summary of all LACP channels.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show etherchannel summary
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show etherchannel summary
```

Output

Figure 23-5: Example output from the show etherchannel summary command

```
% Aggregator po1
%  Admin Key: 0001 - Oper Key 0001
%   Link: port1.0.1 (5001) disabled
%   Link: port1.0.2 (5002) sync: 1
% Aggregator po2
%  Admin Key: 0002 - Oper Key 0002
%   Link: port1.0.6 (5007) disabled
```

show lacp sys-id

Use this command to display the LACP system ID and priority.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show lacp sys-id
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show lacp sys-id
```

Output

Figure 23-6: Example output from the show lacp sys-id command

```
% System Priority: 0x8000 (32768)
% MAC Address: 00-00-cd-24-fd-29
```

show lacp-counter

Use this command to display the packet traffic on all ports of all present LACP aggregators, or a given LACP aggregator.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show lacp-counter [<1-32>]`

Parameter	Description
<1-32>	Channel-group number.

Mode User Exec and Privileged Exec

Example

```
awplus# show lacp-counter 2
```

Output **Figure 23-7: Example output from the show lacp-counter command**

```
% Traffic statistics
Port          LACPDUs         Marker         Pckt err
            Sent    Recv    Sent    Recv    Sent    Recv
% Aggregator po2 (4604)
port1.0.2   0       0       0       0       0       0
```

show port etherchannel

Use this command to show LACP details of the switch port specified.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show port etherchannel <port>`

Parameter	Description
<port>	Name of the switch port to display LACP information about.

Mode User Exec and Privileged Exec

Example

```
awplus# show port etherchannel port1.0.1
```

Output **Figure 23-8: Example output from the show port etherchannel command**

```
% Link: port1.0.1 (5001)
% Aggregator: po1 (4501)
% Receive machine state: Current
% Periodic Transmission machine state: Fast periodic
% Mux machine state: Collecting/Distributing
% Actor Information:                    Partner Information:
%   Selected ............... Selected     Partner Sys Priority ............ 0
%   Physical Admin Key ..... 1           Partner System .. 00-00-00-00-00-00
%   Port Key ............... 5           Port Key ........................ 0
%   Port Priority ......... 32768        Port Priority ................... 0
%   Port Number ..,,,,,,,,,. 5001        Port Number ..................... 0
%   Mode ................... Active      Mode ....................... Passive
%   Timeout ................ Long        Timeout ..................... Short
%   Individual ............. Yes         Individual ..................... Yes
%   Synchronised ........... Yes         Synchronised ................... Yes
%   Collecting ............. Yes         Collecting ..................... Yes
%   Distributing ........... Yes         Distributing ................... Yes
%   Defaulted .............. Yes         Defaulted ...................... Yes
%   Expired ................ No          Expired ........................ No
```

show static-channel-group

Use this command to display all configured static channel groups and their corresponding member ports. Note that a static channel group is the same as a static aggregator.

The command output also shows the thrash limiting status. If thrash limiting is detected and the **thrash limiting** parameter of the **thrash-limiting** command on page 17.57 is set to **vlan disable**, the output will also show the VLANs on which thrashing is detected.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show static-channel-group
```

Mode

User Exec and Privileged Exec

Example

```
awplus#  show static-channel-group
```

Output

Figure 23-9: Example output from the show static-channel-group command

```
% LAG Maximum        : 128
% LAG Static  Maximum: 96
% LAG Dynamic Maximum: 32
% LAG Static  Count  : 2
% LAG Dynamic Count  : 2
% LAG Total   Count  : 4
% Static Aggregator: sa2
% Member:
   port1.0.1
% Static Aggregator: sa3
% Member:
   port1.0.2
```

Related Commands

static-channel-group

static-channel-group

Use this command to create a static channel group, also known as a static aggregator, or add a member port to an existing static channel group.

You can create up to 96 static channel groups (and up to 32 dynamic channel groups).

Use the **no** variant of this command to remove the switch port from the static channel group.

Syntax

```
static-channel-group <static-channel-group-number>
no static-channel-group
```

Parameter	Description
<static-channel-group-number>	<1-96> Static channel group number.

Mode Interface Configuration

Usage This command adds the switch port to the static channel group with the specified channel group number. If the channel group does not exist, it is created, and the port is added to it. The **no** prefix detaches the port from the static channel group. If the port is the last member to be removed, the static channel group is deleted.

All the ports in a channel group must have the same VLAN configuration: they must belong to the same VLANs and have the same tagging status, and can only be operated on as a group.

Once the static channel group has been created, it is treated as a switch port, and can be referred to in other commands that apply to switch ports.

To refer to a static channel group in other static channel group commands, use the channel group number. To specify a static channel group in other commands, prefix the channel group number with **sa**. For example, 'sa2' refers to the static channel group with channel group number 2.

For more on static channel groups, see **"Static Channel Groups" on page 22.3** and **Chapter 22, Link Aggregation Introduction and Configuration**.

Examples To define a static channel group on a switch port, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# static-channel-group 2
```

To reference the pre-defined static channel group 2 as an interface apply the example commands as below:

```
awplus# configure terminal
awplus(config)# interface port1.0.6
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
awplus(config)# interface port.1.0.6
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
awplus(config)# interface sa2
awplus(config-if)#
```

Related Commands **show static-channel-group**

undebug lacp

This command applies the functionality of the **no debug lacp** command on page 23.6.

Chapter 24: Power over Ethernet Introduction

Introduction 24.2
PoE Standards 24.2
PoE (all standards) 24.3
PoE (IEEE 802.3af) 24.3
Enhanced PoE 24.3
PoE+ (IEEE 802.3at) 24.3
Differences Between PoE and PoE+ 24.5
LLDP-MED (TIA-1057) with PoE+ (IEEE 802.3at) 24.5
PoE and PoE+ Applications 24.5
Power Device (PD) Discovery 24.5
Power Classes 24.6
Power through the Cable 24.7
Cable Types 24.8
Static and Automatic Power Allocation 24.8
PoE and PoE+ Implementation 24.9
Power Capacity 24.9
PoE Port Allocation and Distribution 24.9
Power Threshold 24.10
Negotiating Power Requirements 24.10
PoE Port Management 24.10
Powered Device (PD) Detection 24.11
Port Prioritization 24.11
Software Monitoring 24.13
PoE and PoE+ Configuration 24.14
Add a Description for a PoE or PoE+ Port 24.14
Configuring Capacity and Priority on a PoE or PoE+ Port 24.15
Remotely Monitoring Power for all Connected PDs 24.16

Introduction

This chapter provides an introduction to Power over Ethernet (PoE) technology, the PoE standards, PoE devices, and how to configure PoE on your switch. The following x510 Series switches are PoE capable: x510-52GPX, x510-28GPX. For information about the PoE commands available on your switch, see **Chapter 25, Power over Ethernet Commands**.

PoE is a method of supplying power to network devices by utilizing the same cabling used to carry network traffic. PoE is appropriate for devices that have a low power consumption (termed Powered Devices) such as IP phones and security cameras etc. A number of standards have been created to define PoE connectivity. Two PoE standards are presently defined by the Institute of Electrical and Electronics Engineers (IEEE), these are: IEEE 802.3af and IEEE 802.3at.

In addition to the formal PoE methods defined by the IEEE, there are also legacy industry methods for supplying power over Ethernet cabling. For details of legacy support see the command, **power-inline allow-legacy** command on page 25.5.

PoE Standards

PoE is formally defined by the following standards: **formal** (defined by the IEEE), and **de facto** (industry developed):

- IEEE 802.3af Power Ethernet standard
  - « Approved 2003.
  - « Supplies 15.94 W of power of which 12.95 W is available to each powered device.
  - « Superseded by IEEE IEEE802.3at.
  - « Fully supported on the x510-52GPX and x510-28GPX switches

For more information on this standard, refer to **"PoE (IEEE 802.3af)" on page 24.3**.

- Enhanced PoE
  - « Industry standard introduced after the IEEE 802.af. It provides more power (20 W) than defined by IEEE 802.3.af but less than the power (30 W) defined by IEEE 802.3.at.
  - « Compliance with this method is provided on the x510-52GPX and x510-28GPX switches on a best effort basis.
  - « For more information on this standard, refer to **"Enhanced PoE" on page 24.3**.
- IEEE 802.3at Power Ethernet standard (commonly known as PoE+)
  - « Approved 2009.
  - « Supplies 30 W of power of which 25.5 W is available to each powered device.
  - « For more information on this method, refer to **"PoE+ (IEEE 802.3at)" on page 24.3**.
  - « Fully supported on the x510-52GPX and x510-28GPX switches.

PoE (all standards)

The general objective that is common to all PoE methods described in this document is to distribute both data and cable over the same cabling that is used for transmitting Ethernet based data. This eliminates the need for having one set of cables and outlets for data, and another set for power. Also, because the voltage and power requirements are much lower than for mains powered devices, the cabling and installation costs are significantly reduced.

Power Sourcing Equipment (PSE) such as an Ethernet LAN switch or router, supplies power to the cable together with the data. **Powered Devices (PDs)** such as Wireless Access Points or an IP Phones, receive power and data over this same cabling. The PSE employs various methods of power classification (depending on the standard) for detecting compatible PDs from non-compatible devices and will only provide the maximum power limit to compatible PDs, based on their PoE device class. The PSE continuously monitors the PDs and stops providing power when it is no longer requested or it detects an overload or short circuit condition on a port.

PoE (IEEE 802.3af)

The IEEE 802.3at-2003 standard specifies how power is distributed along with data on twisted pair Ethernet LAN cables. The standard specifies that the PSE is able to supply up to 15.4 watts (W) of power (at a nominal 48 VDC), with the full 100 m cable length, is then able to utilize 12.95 W. The difference between these power levels (15.4 -12.98) allows for power loss within the cabling. This figure is approximate and will vary with the cable length and quality. The IEEE 802.3af physical layer classification is a static power allocation based on power bands for power management.

Enhanced PoE

Enhanced was developed prior to IEEE802.3at standard to provide more power the PDs than was currently offered by the old 802.3af standard. Enhanced PoE supplied between 15.4 W and 20 W per port at 48 VDC. Typically, these PD would be used for applications such as building security and video surveillance. Note that IEEE 802.3at standard PoE+ PDs that require 56 VDC cannot use Enhanced PoE PSEs instead of IEEE 802.3at standard PoE+ PSEs. Enhanced PoE PSEs cannot replace IEEE 802.3at standard PSEs when using any PoE+ PDs.

PoE+ (IEEE 802.3at)

The IEEE 802.3at-2009 standard specifies how power is distributed together with data on twisted pair Ethernet LAN cables. PoE+ supplies the higher power required by a new generation of network attached devices. These devices, such as, multiple radio IEEE 802.11n wireless access points, powered pan tilt and zoom IP security cameras, thin clients, door locks, touch screen displays, and video phones frequently require more than the 12.9 W (at a nominal 54 VDC) available under the IEEE 802.3af standard. The IEEE 802.3at specification provides for up to 30 W of power at the PSE, of which 25.5 W is available to the PD.

The standard also requires that PDs support a flexible Layer 2 power classification method using Link Layer Discovery Protocol Media Endpoint Devices (LLDP-MED). The use of LLDP-MED for power classification provides PoE power allocation in steps of 1 watt, along with an ability to reallocate power, for improved power allocation and management between the PSE and PD. For more information see **"LLDP-MED (TIA-1057) with PoE+ (IEEE 802.3at)" on page 24.5**. The IEEE 802.3at specification is backwards compatible with the IEEE 802.3af specification. Devices that support the IEEE 802.3at specification are optimized to operate with IEEE 802.3at PSEs to support dynamic power management. PSEs that support the IEEE 802.3af specification can still interoperate with IEEE 802.3at compliant PDs, providing that the PD can operate using 12.95 W of power. However, these PDs will operate without the dynamic power allocation and management feature.

Differences Between PoE and PoE+

The major differences between the IEEE 802.3af (PoE) and the IEEE 802.3at (PoE+) standards relate to the power that the PDs are allowed to consume, and the ability to dynamically manage the power supplied to each PD. The following table summarizes the major differences in terms of their applied voltages and power ratings.

Standard	Voltage DC at PSE	Cabling	Power Supplied by PSE	Power Available to PD	Nominal Current
IEEE 802.3af	44 V	2 pairs (CAT 3 or better)	15.4 W	12.95 W	350 mA
IEEE 802.3at	(44 to 57) V	2 pairs (CAT 5 or better)	30 W	25.5 W	600 mA

LLDP-MED (TIA-1057) with PoE+ (IEEE 802.3at)

The IEEE 802.1AB standard, Link Layer Discovery Protocol (LLDP) was designed to provide a multi-vendor solution for the discovery of network devices and accurate physical topology of how these devices are connected to one another. LLDP allows network devices to advertise their basic configuration and device capabilities to other network devices on the same LAN.

The IEEE 802.1AB standard was extended by the Telecommunications Industry Association (TIA) to fill the need for multi-vendor VoIP deployments. The TIA created the TIA-1057 standard, Link Layer Discovery Protocol Media Endpoint Devices (LLDP-MED), which allows for Media Endpoint Devices, such as VoIP phones, to exchange configuration information, including Power over Ethernet management. The TIA-1057 standard and the IEEE 802.3at standard provide for the following advanced PoE management capabilities:

- Fine grain PoE power allocation (1 watt granularity instead of wider power class bands)
- Power priority of the PD being supplied power
- Backup power conservation to extend UPS battery life

The IEEE 802.3at standard provides a capability for power re-negotiation with LLDP-MED.

PoE and PoE+ Applications

Products designed to the IEEE 802.3af (PoE) standard and IEEE 802.3at (PoE+) standard provide the benefits of lower installation costs, installation flexibility, and remote power monitoring and device management. Products supporting IEEE 802.3at can use higher power levels, along with dynamic power management when using LLDP-MED to exchange configuration data.

Power Device (PD) Discovery

The first step for PSE equipment(your x510 switch, for example) is to determine whether a device plugged into a port is a valid Powered Device (PD). If it is, it will require power as well as network communication through the attached LAN cable.

The IEEE 802.3af-2003 and IEEE 802.3at-2009 standards for device detection involves applying a DC voltage between the transmit and receive wire pairs, and measuring the received current.

A PSE will expect to see approximately 25K Ohm resistance and 150nF capacitance between the transmit and receive wire pairs for the device to be considered a valid PD. A range around these values is specified in the IEEE 802.3af and IEEE 802.3at Power Ethernet standards.

The PSE will check for the presence of PDs on connected ports at regular intervals, so that power can be removed when a PD is no longer connected. Legacy (pre-IEEE 802.3af Power Ethernet standard) PDs are also detected by the PSE by default. See **"power-inline allow-legacy" on page 25.5**.

Power Classes

Once a PD is discovered, PSE initiates a PD classification test by applying a DC voltage to the port. If the PD supports optional power classification it will apply a load to the line to indicate to the PSE the classification the device requires.

Since PDs may require differing power ranges, the IEEE 802.3af and IEEE 802.3at Power Ethernet standards classifies PDs according to their power consumption. By providing the PSE with its power range, the PD allows the PSE to supply power with greater efficiency. The power classes as outlined by IEEE 802.3af and IEEE 802.3at are as follows showing the different PD classes and the PSE power output for each corresponding PD power range:

PD Class	Power Available at PD	Power Supplied from PSE
0	0.44 W to 12.95 W	15.4 W
1	0.44 W to 3.84 W	4.0 W
2	3.84 W to 6.49 W	7.0 W
3	6.49 W to 12.95 W	15.4 W
4	12.95 W to 25.5 W	30 W

Once the PSE has detected the PDs IEEE 802.3af or IEEE 802.3at power class, it can manage the power allocation by subtracting the PDs class maximum value from the overall power budget. This allows for control and management of power allocation when there is not enough power available from the PSE to supply maximum power to all ports. Any unclassified PD is considered to be a class 0 device.

To view the PD class that has been configured for each PoE port, apply the following command:

```
awplus# show power-inline
```

Typical Values for PD Power Consumption

The IEEE 802.3af standard specifies the delivery of up to 15.4 watts (W) per port to PoE devices. This enables a variety of possible devices to make use of the available power. The maximum power consumed by a PD, as specified by the IEEE 802.3af standard, is 12.95 W. The system provides the 'extra' power (up to 15.4 W) to compensate for losses in the cable. Some common PoE device power requirements are:

PoE Device	PoE Power Requirement
IP phone	3 W-6 W
Wireless access point	4 W-11 W
IP security camera	5 W-12 W

The IEEE 802.3at standard supports delivery of up to 30 W per port that may be used to deliver power to PoE+ devices. This allows a variety of possible devices to make use of the available power. The maximum power consumed by a PD, as specified by the IEEE 802.3at standard, is 25.5 W. The system provides the 'extra' power (up to 30 W) to compensate for line loss. Some common PoE+ device power requirements are:

PoE+ Device	PoE+ Power Requirement
Wireless Access Point (with LLDP-MED support)	12 W-24 W
Pan Tilt and Zoom powered IP security camera	12 W-24 W

Refer to the LLDP chapters **Chapter 96, LLDP, LLDP-MED and Voice VLAN Introduction and Configuration** and **Chapter 97, LLDP Commands** for information about power monitoring at the PD.

Power through the Cable

10/100BASE-TX Endpoint Mode

The IEEE 802.3af and IEEE 802.3at standards describe two methods for applying PoE over twisted pair cabling are termed “alternatives A and B”. Alternative A applies power to the data carrying cable pairs (using pins 1-2 and 3-6). Alternative B applies the power to the spare cable pairs (using pins 4-5 and 7-8).

The x510 series switches use “alternative A” to apply power to its PDs. An IEEE compliant PD should be able to receive PoE using either of the two wiring methods.

1000BASE-T Endpoint Mode

An amendment to the IEEE 802.3at (2008) standard defines PoE cable connections for data transmission at 1 GHz. Although data is carried over all four cable pairs, the same cable pinning is used for PoE. i.e. alternative A applies power to the cable using pins 1-2 and 3-6, and alternative B applies the power to the cable using pins 4-5 and 7-8.

Cable Types

Although the IEEE standards 802.3af and 802.3at indicate minimum cable types for each standard version; using cables of a higher rating will reduce the cable resistance, allowing more power to be provided from the PSE to the PD. Also the power negotiation process that takes place between the PD and the PSE takes no account of the cable type that connects them, therefore the cabling used should be rated to meet the highest power that your PSE is able to supply

A further factor is that network operation at 1 Gbps places higher demands on the cabling type used. These demands are increased when power is also carried over these cables.

For more information on twisted-pair cable selection for PoE, see the **Allied Telesis x510 Series Installation Guide**.

Static and Automatic Power Allocation

When configuring PoE on your switch, you can either allow each port to auto-negotiate its power requirement, based on the power class of its connected PD, or you can statically allocate fixed power levels to each port.

Where dynamic PoE power assignment is used, the total power drawn from your switch's power supply will be the total of the individual power requirements of each port. See **"power-inline max" on page 25.8**

PoE and PoE+ Implementation

The following x510 series switches support PoE:

- x510-52GPX
- x510-28GPX

This section explains how to implement PoE on these switches.

Power Capacity

The following information is provided as a guide, and we recommend that you consult the appropriate Installation Guide for your particular switch for more detailed information.

The x510 series switches are supplied with two internal power supplies (PSUs). For resiliency, each of these PSUs has its own external mains connection. When ports are supplying PoE, the dual PSU facility enables you to run the switch in one of two modes:

- standard (Redundant) mode
- boost mode

In standard (Redundant) mode you apply power to both PSUs but restrict the power demands of your PDs to be within the capabilities of a single power supply.

In boost mode you employ both PSUs, and utilize more power than is available from a single PSU. However, if power from one of the PSUs is lost, the result will be a loss of PoE capability to a number of ports - the exact number will depend on your particular port configuration.

Note Note that you can preserve the operation of a number of essential PoE ports by using the command, **"power-inline priority" on page 25.10**. For more details of using this feature, see **"Port Prioritization" on page 24.11**.

The specific power capacity available depends on the model purchased, x510-52GPX or x510-28GPX. In this respect, the following information is provided as a guide, and we recommend that you consult your switch's hardware reference for more accurate information.

PoE Port Allocation and Distribution

The total power available for PoE is dependent on the capacity of the power supply. How this power is distributed across the available PoE ports depends on the PoE configuration applied to the switch, particularly how you set the **power-inline priority** and **power-inline max** commands.

The x510-28GPX and the x510-52GPX model switches are both Layer 3 PoE+ Gigabit Ethernet switches with SFP and SFP+ support.

Power Threshold

The switch can be configured to send a Simple Network Management Protocol (SNMP) trap to your management workstation and records an entry in the event log whenever the total power requirements of the powered devices exceed the specified percentage of the total maximum power available on the switch. With the default setting of 80% applied, the switch sends an SNMP trap when the PoE devices require more than 80% of the maximum available power on the switch.

To adjust the threshold, use the command:

```
awplus(config)# power-inline usage-threshold <1-99>
```

For your management workstation to receive traps from your switches, you must configure SNMP on the switch by specifying the IP address of the workstation. The management workstation will also record an entry in the event log whenever power consumption of the switch has returned to a value that is less than the power limit threshold.

To set the SNMP traps (notifications) for PoE, use the command:

```
awplus(config)# snmp-server enable trap power-inline
```

See **Chapter 93, SNMP Introduction** for information about configuring SNMP traps for PoE. See **Chapter 94, SNMP Commands** for command examples to configure SNMP traps for PoE.

Negotiating Power Requirements

When configuring PoE on your switch, you can either allow each port to auto-negotiate its power requirement, based on the power class of its connected PD, or you can statically configure each port to have a fixed power allocation.

Where dynamic PoE power assignment is used, the total power assigned from your switch's power supply will be the total of the individual power requirements assigned to each port.

To enable your PoE ports to dynamically allocate their power, set the **"power-inline max" on page 25.8** to its default by entering, no power-inline max.

PoE Port Management

PoE is enabled by default on all non-SFP (or SFP+) RJ-45 ports. You can connect either a powered or non-powered device to a PoE-enabled port without having to re-configure the port. This is because PD detection is carried out before any power is supplied to the connected device.

PoE can be administratively enabled or disabled on each port using the **power-inline enable** command in Interface Configuration mode. To disable PoE on a selected port, use the command:

```
awplus(config-if)# no power-inline enable
```

Powered Device (PD) Detection

Your switch applies two methods to detect the connection of a PD. The first method applies the resistance and capacitance methods defined in the IEEE standards, see **"Power Device (PD) Discovery" on page 24.5**. The second method is applied to detect the connection of legacy PDs. This method involves measuring for a large capacitance value. The IEEE method is tried first, and if this fails to detect a PD, the second method is applied.

By default, legacy PD detection is enabled on all ports. To disable legacy PD detection, use the command:

```
awplus(config)# no power-inline allow-legacy
```

The switch applies its PD detection process in real time to all PoE enabled ports. It will not supply power to any PoE enabled port unless it detects the connection of a valid PD.

Port Prioritization

Port prioritization enables you to assign ports to be one of the following three priority levels:

- Critical
- High
- Low

Where the power required collectively by the Powered Devices (PDs) is greater than the PSUs are able to supply, these priority levels will be used to sequentially remove power from the PDs in an order of their importance.

Critical The highest priority level. Ports set to this level are guaranteed power before any ports assigned to the other two priority levels. Ports assigned to the other priority levels receive power only if all the Critical ports are receiving power. Your most critical powered devices should be assigned to this level. If there is not enough power to support all the ports set to the Critical priority level, power is provided to the ports based on port number, in ascending order.

High The second highest level. Ports set to this level receive power only if all the ports set to the Critical level are already receiving power. If there is not enough power to support all of the ports set to the High priority level, power is provided to the ports based on port number, in ascending order.

Low The lowest priority level. This is the default setting. Ports set to this level only receive power if all the ports assigned to the other two levels are already receiving power. As with the other levels, if there is not enough power to support all of the ports set to the Low priority level, power is provided to the ports based on port number, in ascending order

If power needs to be removed from some of the PoE ports, where for example, one of the power supplies is disconnected; power will be removed from these ports in the order Low, High, and Critical. In addition, within each of these priority categories, lower numbered ports will be given higher priority than higher numbered ports; i.e. the lower the port number, the higher its PoE priority (within its particular category).

You can set the port priority by using the command:

```
awplus# power-inline priority
```

For more details on using this command see **"power-inline priority" on page 25.10**.

Note

Power allocation is dynamic. Ports supplying power may stop powering a PD if the switch's power capacity has reached maximum usage and new PD's are connected to ports with a higher priority, which become active.

To ensure continued operation of a PD if the power resources of the switch are exceeded you should install a PD to a lower numbered PoE port with the Critical priority level configured.

Software Monitoring

There are four PoE **show** commands available that return information about the PoE settings on your switch.

The **show power-inline** command displays the power threshold set, a power usage percentage, and power consumed by each switch port.

```
awplus# show power-inline
```

The **show power-inline counters** command displays PoE event counters from the PoE MIB (RFC 3621).

```
awplus# show power-inline counters
```

The **show power-inline interface** command displays a summary of all PoE information, including power limit, power consumed, and power class.

```
awplus# show power-inline interface
```

The **show power-inline interface detail** command displays all PoE information, including power limit, power consumed, and power class.

```
awplus# show power-inline interface detail
```

You can also specify an individual PoE port, a range of PoE ports, or a selection of PoE ports with the **show power-inline interface detail** command when using the <*port-list*> option, as shown below for a PoE port, a selection of PoE ports, and a range of PoE ports:

```
awplus# show power-inline interface port1.0.2 detail

awplus# show power-inline interface port1.0.2,port1.0.4
        detail

awplus# show power-inline interface port1.0.2-port1.0.4
        detail
```

PoE and PoE+ Configuration

This section is based around PoE configuration tasks for the Allied Telesis x510-28GPX and x510-52GPX switches running the AlliedWare Plus™ Operating System.

Add a Description for a PoE or PoE+ Port

You can add a description (typically the device type) for a PoE port, which the switch will display in certain Show commands. Knowing the type of PD is useful when inspecting PD Class power usage. The description entered will appear in the following commands under Device, or Powered Device Type, for each PoE or PoE+ port:

show power-inline interface command on page 25.20 and,

show power-inline interface detail command on page 25.22.

In the following example a description is added for the PoE port listed as **port1.0.2** to display the words "**Desk Phone"** in the show output of the commands mentioned above.

Command	Description
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `configure interface port1.0.2`	Specify the PoE or PoE+ port to be configured and enter Interface mode.
`awplus(config-if)#` `power-inline description Wireless Access Point # 1`	The description "Desk Phone" will be displayed in all PoE show command output for `port1.0.2`.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `exit`	Return to Privileged Exec mode.
`awplus#` `show power-inline interface port1.0.2`	Display the PoE status for `port1.0.2` to confirm that your PoE configuration on the PSE has been successful. If a PD is connected to the configured PoE port then power consumption as well as power allocation values will display.
`awplus#` `copy running-config startup-config`	Save your running-config to the startup-config to keep your PoE configuration after a switch restart or reboot.

Configuring Capacity and Priority on a PoE or PoE+ Port

The following commands set a higher priority and a lower maximum power for a PoE or PoE+ port. This prevents high powered PDs from being connected to a PoE or PoE+ port reserved for low powered PDs. Follow the configuration table below to configure **port1.0.2**.

Command	Description
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `configure interface port1.0.2`	Specify the PoE or PoE+ port to be configured and enter Interface mode.
`awplus(config-if)#` `power-inline priority high`	Specify a higher priority for the PoE or PoE+ port than the default low setting.
`awplus(config-if)#` `power-inline max 4000`	Specify the lowest available power that the PSE can supply to the PD: 4000 mW.
`awplus(config-if)#` `exit`	Return to Global Configuration mode.
`awplus(config)#` `exit`	Return to Privileged Exec mode.
`awplus#` `show power-inline interface port1.0.2`	Display the PoE status for `port1.0.2` to confirm that your PoE configuration on the PSE has been successful. If a PD is connected to the configured PoE port then power consumption as well as power allocation values will display.
`awplus#` `copy running-config startup-config`	Save your running-config to the startup-config to keep your PoE configuration after a switch restart or reboot.

Remotely Monitoring Power for all Connected PDs

By using the **power-inline usage-threshold** command and the **snmp-server enable trap** commands together you can remotely monitor PD power requests on the PSE.

Note that you will need to configure SNMP first for this. For more information on configuring SNMP, see the following chapters in your switch's software reference:

- **Chapter 93, SNMP Introduction**
- **Chapter 94, SNMP Commands**
- **Chapter 95, SNMP MIBs**

For example, if the PD is a Class 0 (default class) or a Class 3 (15400 mW) PD then the PSE budgets 15400 mW for the PD regardless of the actual amount of power needed by the PD.

The following procedure allows you to remotely monitor power usage for all connected PDs. Follow the configuration table to configure the PSE.

Command	Description
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `service power-inline`	Enable PoE globally for the PSE. This will also enable PoE globally for all PoE ports on all connected stacked switches.
`awplus(config)#` `snmp-server enable trap power-inline`	Configure SNMP notification so an SNMP trap is sent when the power usage threshold is exceeded to trigger an alarm.
`awplus(config)#` `power-inline usage-threshold 75`	Specify SNMP notifications are generated when the power supplied exceeds 75% of the nominal PSE power available.
`awplus(config)#` `exit`	Return to Privileged Exec mode.
`awplus#` `show power-inline`	Display the PoE status for all PoE ports on the PSE. The PD Class, power consumption, and power allocated per PoE port displays for all PoE ports on the PSE.
`awplus#` `copy running-config startup-config`	Save your running-config to the startup-config to keep your PoE configuration after a switch restart or reboot.

Chapter 25: Power over Ethernet Commands

Introduction 25.2
Command List 25.2
clear power-inline counters interface 25.3
debug power-inline 25.4
power-inline allow-legacy 25.5
power-inline description 25.6
power-inline enable 25.7
power-inline max 25.8
power-inline priority 25.10
power-inline usage-threshold 25.12
service power-inline 25.13
show debugging power-inline 25.14
show power-inline 25.15
show power-inline counters 25.18
show power-inline interface 25.20
show power-inline interface detail 25.22

Introduction

Power over Ethernet (PoE) is a technology allowing devices such as IP phones to receive power over existing LAN cabling.

PoE is configured using the commands in this chapter. Note the Power Sourcing Equipment (PSE) referred to throughout this chapter is an Allied Telesis PoE switch running the AlliedWare Plus™ Operating System, supporting the IEEE 802.3af and IEEE 802.3at Power Ethernet standards. The Powered Device (PD) referred to throughout this chapter is a PoE or PoE+ powered device, such as an IP phone or a Wireless Access Point (WAP).

The commands in this chapter are available on the Allied Telesis x510-52GPX and x510-28GPX switches:

Command List

This chapter contains an alphabetical list of commands used to configure Power over Ethernet (PoE). Each command contains a functional description and shows examples of configuration and output screens for show commands. These commands are only supported on PoE capable ports. An error message will display on the console if you enter a PoE command on a port that does not support PoE. The following chapters offer further information for configuring PoE on Allied Telesis switches.

- **Chapter 24, Power over Ethernet Introduction** for introductory information about PoE and how to configure PoE on Allied Telesis switches.
- **Chapter 95, SNMP MIBs** for information about which PoE MIB objects are supported.
- **Chapter 93, SNMP Introduction** for information about SNMP traps.
- **Chapter 94, SNMP Commands** for SNMP command descriptions used when configuring SNMP traps for PoE.

clear power-inline counters interface

This command will clear the counters from a specified port, a range of ports, or all ports on the Power Sourcing Equipment (PSE). If no ports are entered then PoE counters for all ports are cleared. It will also clear all Power over Ethernet (PoE) counters supported by the Power Ethernet MIB (RFC 3621).

Syntax `clear power-inline counters interface [<port-list>]`

Parameter	Description
<port-list>	Selects the port or ports whose counters are to be cleared.

Mode Privileged Exec

Usage The PoE counters are displayed with the **show power-inline counters** command.

Examples To clear the PoE counters for `port1.0.2` only, use the following command:

```
awplus# clear power-inline counters interface port1.0.2
```

To clear the PoE counters for `port1.0.1` through `port1.0.10`, use the following command:

```
awplus# clear power-inline counters interface port1.0.1-
port1.0.10
```

To clear the PoE counters for all ports, use the following command:

```
awplus# clear power-inline counters interface
```

Validation Commands **show power-inline counters**

debug power-inline

This command enables debugging display for messages that are specific to Power over Ethernet (PoE).

Use the **no** variant of this command to disable the specified PoE debugging messages.

Syntax

```
debug power-inline [all|event|info|power]
no debug power-inline [all|event|info|power]
```

Parameter	Description
`all`	Displays all (`event`, `info`, `nsm`, `power`) debug messages.
`event`	Displays event debug information, showing any error conditions that may occur during PoE operation.
`info`	Displays informational level debug information, showing high-level essential debugging, such as information about message types.
`power`	Displays power management debug information.

Default No debug messages are enabled by default.

Mode Privileged Exec

Usage Use the **terminal monitor** command to display PoE debug messages on the console.

Use the **show debugging power-inline** command to show the PoE debug configuration.

Examples To enable PoE debugging and start the display of PoE `event` and `info` debug messages on the console, use the following commands:

```
awplus# terminal monitor
awplus# debug power-inline event info
```

To enable PoE debugging and start the display of all PoE debugging messages on the console, use the following commands:

```
awplus# terminal monitor
awplus# debug power-inline all
```

To disable PoE debugging and stop the display of PoE `event` and `info` debug messages on the console, use the following command:

```
awplus# no debug power-inline event info
```

To disable all PoE debugging and stop the display of any PoE debugging messages on the console, use the following command:

```
awplus# no debug power-inline all
```

Validation Commands **show debugging power-inline**

Related Commands **terminal monitor**

power-inline allow-legacy

This command enables detection of pre-IEEE 802.3af Power Ethernet standard legacy Powered Devices (PDs).

The no variant of this command disables detection of pre-IEEE 802.3af Power Ethernet standard legacy Powered Devices (PDs).

Syntax

```
power-inline allow-legacy
no power-inline allow-legacy
```

Default Detection of legacy PDs is enabled on all ports on the Power Sourcing Equipment (PSE).

Mode Global Configuration

Examples To disable detection of legacy PDs, use the following commands:

```
awplus# configure terminal
awplus(config)# no power-inline allow-legacy
```

To enable detection of legacy PDs, use the following commands:

```
awplus# configure terminal
awplus(config)# power-inline allow-legacy
```

Validation Commands **show power-inline**
show running-config power-inline

power-inline description

This command adds a description for a Powered Device (PD) connected to a PoE port.

The **no** variant of this command clears a previously entered description for a connected PD, resetting the PD description to the default (null).

Syntax

```
power-inline description <pd-description>
no power-inline description
```

Parameter	Description
<pd-description>	Description of the PD connected to the PoE capable port (with a maximum 256 character string limit per PD description).

Default No description for a connected PD is set by default.

Mode Interface Configuration

Usage Select a PoE port, a list of PoE ports, or a range of PoE ports with the preceding **interface (to configure)** command. If you specify a range or list of ports they must all be PoE capable ports.

Examples To add the description `Desk Phone` for a connected PD on `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# power-inline description Desk Phone
```

To clear the description as added above for the connected PD on `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no power-inline description
```

Validation Commands **show power-inline interface**
show running-config power-inline

power-inline enable

This command enables Power over Ethernet (PoE) to detect a connected Powered Device (PD) and supply power from the Power Sourcing Equipment (PSE).

The **no** variant of this command disables PoE functionality on the selected PoE port(s). No power is supplied to a connected PD after PoE is disabled on the selected PoE port(s).

Syntax

```
power-inline enable
no power-inline enable
```

Default PoE is enabled by default on all ports on the PSE.

Mode Interface Configuration

Usage In a stack of x510 Series switches this command is supported on all PoE capable ports.

Select a PoE port, a list of PoE ports, or a range of PoE ports from the preceding **interface (to configure)** command. If you specify a range or list of ports they must all be PoE capable ports.

No PoE log messages are generated for specified PoE port(s) after PoE is disabled. The disabled PoE port(s) still provide Ethernet connectivity after PoE is disabled.

Examples To disable PoE on ports `port1.0.1` to `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.4
awplus(config-if)# no power-inline enable
```

To enable PoE on ports `port1.0.1` to `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.4
awplus(config-if)# power-inline enable
```

Validation Commands

show power-inline
show power-inline interface
show power-inline interface detail
show running-config power-inline

power-inline max

This command sets the "maximum" power allocated to a Power over an Ethernet (PoE and PoE+) port. The amount of power actually supplied to the port depends on the power requirements of the connected PD. It is also a function of the total PoE power loading on the switch and the PoE priority set for the port by the **power-inline priority** command. However this command (power-inline max) does apply a "maximum" value to the power that the port is able to supply.

Note Note that the value set by this command will be the figure the switch will use when apportioning the power budget for its ports. For example, if 15.4 W is assigned to a port whose PD only consumes 5 W, the switch will reserve the full 15.4 W for this port when determining its total power PoE power requirement.

Note that the value set by this command will be the figure the switch will use when apportioning the power budget for its ports. For example, if 15.4 W is assigned to a port whose PD only consumes 5 W, the switch will reserve the full 15.4 W for this port when determining its total power PoE power requirement.

The **no** variant of this command sets the maximum power supplied to a PoE port to the default, which is set to the maximum power limit for the class of the connected Powered Device (PD).

Syntax

```
power-inline max <4000-30000>
no power-inline max
```

Parameter	Description
<4000-30000>	The maximum powerallocated to a PoE port in milliwatts (mW).

Default The Power Sourcing Equipment (PSE) supplies the maximum power limit for the class of the PD connected to the port by default.

Note Power limits for all classes of PDs are listed in **"Power Classes" on page 24.6**. See **Chapter 24, Power over Ethernet Introduction** for further PoE information.

Mode Interface Configuration

Usage In a stack of x510 Series switches this command is supported on all PoE capable ports.

Select a PoE port, a list of PoE ports, or a range of PoE ports with the preceding **interface (to configure)** command. If you specify a range or list of ports they must all be PoE capable ports.

If you select a range of PoE ports in Interface Configuration mode before issuing this command, then each port in the range selected will have the same maximum power value configured. If the PoE port attempts to draw more than the maximum power, this is logged and all power is removed. Note that the value entered is rounded up to the next value supported by the hardware.

See the actual value used, as shown after command entry, in the sample console output below:

```
awplus#configure terminal
awplus(config)#interface port1.0.1
awplus(config-if)#power-line max 5300
% The maximum power has been rounded to 5450mW in hardware.
```

Refer to **Chapter 96, LLDP, LLDP-MED and Voice VLAN Introduction and Configuration** and **Chapter 97, LLDP Commands** for information about power monitoring at the PD.

Note the difference in power supplied from the PSE to the power available at the PD due to line loss.

The **"Power Classes" on page 24.6** shows the difference between the power supplied from the PSE and the power available at the PD.

Examples

To set the maximum power supplied to ports in the range `1.0.2 to 1.0.4` to 6450mW per port, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.4
awplus(config-if)# power-inline max 6450
```

To set the maximum power supplied to `port1.0.2`, to `6450` mW, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# power-inline max 6450
```

To clear the user-configured maximum power supplied to `port1.0.2`, and revert to using the default maximum power of 30000 mW, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no power-inline max
```

Validation Commands

show power-inline interface
show running-config power-inline

power-inline priority

This command sets the Power over Ethernet (PoE) priority level of a PoE port to one of three available priority levels:

- low
- high
- critical

The **no** variant of this command restores the PoE port priority to the default (low).

Syntax

```
power-inline priority {low|high|critical}

no power-inline priority
```

Parameter	Description
`low`	The lowest priority for a PoE enabled port (default). PoE ports set to `low` only receive power if all the PoE ports assigned to the other two levels are already receiving power.
`high`	The second highest priority for a PoE enabled port. PoE ports set to `high` receive power only if all the ports set to `critical` are already receiving power.
`critical`	The highest priority for a PoE enabled port. PoE ports set to `critical` are guaranteed power before any ports assigned to the other two priority levels. Ports assigned to the other priority levels receive power only if all Critical ports are receiving power.

Default The default priority is `low` for all PoE ports on the Power Sourcing Equipment (PSE).

Mode Interface Configuration

Usage This command is supported on all PoE capable ports, whether operating as a stand-alone switch, or within a VCStack.

Select a PoE port, a list of PoE ports, or a range of PoE ports with the preceding **interface (to configure)** command. If you specify a range or list of ports they must all be PoE capable ports.

PoE ports with higher priorities are given power before PoE ports with lower priorities. If the priorities for two PoE ports are the same then the lower numbered PoE port is given power before the higher numbered PoE port.

See **"Port Prioritization" on page 24.11** for further information about PoE priority.

Examples To set the priority level to `high` for `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# power-inline priority high
```

To reset the priority level to the default for `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no power-inline priority
```

Validation Commands
show power-inline
show power-inline interface
show running-config power-inline

Related Commands **power-inline usage-threshold**

power-inline usage-threshold

This command sets the level at which the Power Sourcing Equipment (PSE) will issue a message that the power supplied to all Powered Devices (PDs) has reached a critical level of the nominal power rating for the PSE. The level is set as a percentage of total available power.

The **no** variant of this command resets the notification usage-threshold to the default (80% of the nominal power rating of the PSE).

Syntax

```
power-inline usage-threshold <1-99>

no power-inline usage-threshold
```

Parameter	Description
<1-99>	The usage-threshold percentage configured with this command.

Default The default power usage threshold is 80% of the nominal power rating of the PSE.

Mode Global Configuration

Usage Use the **snmp-server enable trap** command on page 94.18 to configure SNMP notification. An SNMP notification is sent when the usage-threshold, as configured in the example, is exceeded.

Examples To generate SNMP notifications when power supplied exceeds 70% of the nominal PSE power, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap power-inline
awplus(config)# power-inline usage-threshold 70
```

To reset the notification threshold to the default (80% of the nominal PSE power rating), use the following commands:

```
awplus# configure terminal
awplus(config)# no power-inline usage-threshold
```

Validation Commands **show power-inline interface**
show running-config power-inline

Related Commands **snmp-server enable trap**

service power-inline

This command enables Power over Ethernet (PoE) globally on the Power Sourcing Equipment (PSE) for all PoE ports.

Syntax

```
service power-inline
no service power-inline
```

Default PoE functionality is enabled by default on the PSE.

Mode Global Configuration

Usage In a stack, issuing this command enables PoE globally for all PoE ports.

In a stack configuration, only stack members containing PoE hardware will have PoE enabled by default in software.

Examples To disable PoE on the PSE, use the following commands:

```
awplus# configure terminal
awplus(config)# no service power-inline
```

To re-enable PoE on the PSE, if PoE has been disabled, use the following commands:

```
awplus# configure terminal
awplus(config)# service power-inline
```

Validation Commands **show power-inline**
show running-config power-inline

show debugging power-inline

This command displays Power over Ethernet (PoE) debug settings.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging power-inline`

Mode User Exec and Privileged Exec

Example To display PoE debug settings, use the following command:

```
awplus# show debugging power-inline
```

Output **Figure 25-1: Example output from the show debugging power-inline command**

```
awplus#show debugging power-inline
PoE Debugging status:
PoE Informational debugging is disabled
PoE Event debugging is disabled
PoE Power Management debugging is disabled
PoE NSM debugging is enabled
```

Related Commands **debug power-inline**
terminal monitor

show power-inline

This command displays the Power over Ethernet (PoE) status for all ports on the Power Sourcing Equipment (PSE).

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show power-inline`

Mode User Exec and Privileged Exec

Example To display the PoE status for all ports on the PSE, use the following command:

```
awplus# show power-inline
```

Output **Figure 25-2: Example output from the show power-inline command**

```
awplus#show power-inline
PoE Status:

Stack Member 2
 Nominal Power: 370W
 Power Allocated: 246W
 Actual Power Consumption: 151W
 Operational Status: On
 Power Usage Threshold: 80% (296W)

PoE Interface:
Interface  Admin   Pri  Oper    Power   Device    Class  Max(mW)
port2.0.1  Enabled Low  Powered  3840   n/a          1    4000 [C]
port2.0.2  Enabled High Powered  6720   n/a          2    7000 [C]
port2.0.3  Enabled Low  Powered 14784   n/a          3   15400 [C]
port2.0.4  Enabled Crit Powered 14784   n/a          3   15400 [C]
port2.0.5  Enabled Crit Powered  3840   n/a          1    4000 [C]
port2.0.6  Enabled High Powered  6720   n/a          2    7000 [C]
port2.0.7  Enabled Low  Powered 14784   n/a          3   15400 [C]
```

Table 25-1: Parameters in the show power-inline command output

Parameter	Description
`Nominal Power`	The nominal power available on the switch in watts (W).
`Power Allocated`	The current power allocated in watts (W) that is available to be drawn by any connected Powered Devices (PDs). This is updated every 5 seconds.
`Actual Power Consumption`	The current power consumption in watts (W) drawn by all connected Powered Devices (PDs). This is updated every 5 seconds.
`Operational Status`	The operational status of the PSU hardware on the PSE when this command was issued: ■ **On** if the PSU as installed inside the PSE is switched on. ■ **Off** when the PSU in the PSE is switched off (an RPS may be connected to the PSE to power PoE instead of the PSU). ■ **Fault** when there is an issue with the PSE PSU hardware.
`Power Usage Threshold (%)`	The configured SNMP trap / log threshold for the PSE, as configured from a **power-inline usage-threshold** command.
`Interface`	The PoE port(s) in the format `portx.y.z`, where `x` is the device number, `y` is the module number within the device, and `z` is the PoE port number within the module.
`Admin`	The administrative state of PoE on a PoE port, either **Enabled** or **Disabled**.
`Pri`	The current PoE priorities for PoE ports on the PSE, as configured from a **power-inline priority** command: ■ **Low** displays when the `low` parameter is issued. The lowest priority for a PoE enabled port (default). ■ **High** displays when the `high` parameter is issued. The second highest priority for a PoE enabled port. ■ **Crit** displays when the `critical` parameter is issued. The highest priority for a PoE enabled port.
`Oper`	The current PSE PoE port state when this command was issued: ■ **Powered** displays when there is a PD connected and power is being supplied from the PSE. ■ **Disabled** displays when supplying power would make the PSE go over the power budget. ■ **Off** displays when PoE has been disabled for the PoE port. **Fault** displays when a PSE goes over its power allocation.
`Power`	The power consumption in milliwatts (mW) for the PoE port when this command was entered.
`Device`	The description of the connected PD device if a description has been added with the **power-inline description** command. No description is shown for PDs not configured with the **power-inline description** command.

Table 25-1: Parameters in the show power-inline command output (cont.)

Parameter	Description
`Class`	The class of the connected PD, if power is being supplied to the PD from the PSE. See the **Power over Ethernet Introduction** chapter for further information about PD classes and the power levels assigned per class.
`Max (mW)`	The power in milliwatts (mW) allocated for the PoE port. Additionally, note the following as displayed per PoE port: ■ **[U]** if the power limit for a port was user configured (with the **power-inline max** command). ■ **[L]** if the power limit for a port was supplied by LLDP. ■ **[C]** if the power limit for a port was supplied by the PD class.

Related Commands

show power-inline counters
show power-inline interface

show power-inline counters

This command displays Power over Ethernet (PoE) event counters for ports on the Power Sourcing Equipment (PSE). The PoE event counters displayed can also be accessed by objects in the PoE MIB (RFC 3621). See **Chapter 95, SNMP MIBs** for information about which PoE MIB objects are supported.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show power-inline counters [<port-list>]
```

Parameter	Description
<port-list>	Enter the PoE port(s) to display all PoE event counters for them.

Mode User Exec and Privileged Exec

Usage To display all PoE event counters for all PoE ports on the PSE, do not enter the optional interface parameter.

Examples To display all PoE event counters for all PoE ports on the PSE, use the command:

```
awplus# show power-inline counters
```

To display the PoE event counters for the port range `1.0.1` to `1.0.3`, use the command:

```
awplus# show power-inline counters interface port1.0.1-1.0.3
```

Output **Figure 25-3: Example output from the show power-inline counters command**

```
awplus#show power-inline counters interface port1.0.4-port1.0.12
PoE Counters:
Interface   MPSAbsent Overload Short Invalid Denied
port1.0.4   0         0        0     0       0
port1.0.5   0         0        0     0       0
port1.0.6   0         0        0     0       0
port1.0.7   0         0        0     0       0
port1.0.8   0         0        0     0       0
port1.0.9   0         0        0     0       0
port1.0.10  0         0        0     0       0
port1.0.11  0         0        0     0       0
port1.0.12  0         0        0     0       0
```

Table 25-2: Parameters in the show power-inline counters command output

Parameter	Description
`Interface`	The PoE port(s) in the format `portx.y.z`, where `x` is the device number, `y` is the module number within the device, and `z` is the PoE port number within the module.
`MPSAbsent`	The number of instances when the PoE MPS (Maintain Power Signature) signal has been lost. The PoE MPS signal is lost when a PD is disconnected from the PSE. Also increments `pethPsePortMPSAbsentCounter` in the PoE MIB.
`Overload`	The number of instances when a PD exceeds its configured power limit (as configured by the **power-inline max** command). Also increments `pethPsePortOverLoadCounter` in the PoE MIB.
`Short`	The number of short circuits that have happened with a PD. Also increments `pethPsePortShortCounter` in the PoE MIB.
`Invalid`	The number of times a PD with an Invalid Signature (where the PD has an open or short circuit, or is a legacy PD) is detected. Also increments `pethPseInvalidSignatureCounter` in the PoE MIB.
`Denied`	The number of times a PD has been refused power due to power budget limitations for the PSE. Also increments `pethPsePortPowerDeniedCounter` in the PoE MIB.

Related Commands

clear power-inline counters interface
show power-inline
show power-inline interface

show power-inline interface

This command displays a summary of Power over Ethernet (PoE) information for specified ports. If no ports are specified then PoE information is displayed for all ports on the Power Sourcing Equipment (PSE).

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show power-inline interface [<port-list>]`

Parameter	Description
<port-list>	Enter the PoE port(s) to display PoE specific information in the show output.

Mode User Exec and Privileged Exec

Usage To display PoE information for all PoE ports on the PSE, do not specify any ports.

Example To display the PoE port specific information for all PoE ports on the switch, use the following command:

```
awplus# show power-inline interface
```

To display the PoE port specific information for the port range `1.0.1` to `1.0.4`, use the following command:

```
awplus# show power-inline interface port1.0.1-port1.0.4
```

Output **Figure 25-4: Example output from the show power-inline interface command**

```
awplus#show power-inline interface port1.0.1-port1.0.4
Interface  Admin    Pri  Oper      Power Device      Class Max(mW)
port1.0.1  Disabled Low  Disabled      0 n/a         n/a   n/a
port1.0.2  Enabled  High Powered    3840 Desk Phone  1     5000 [U]
port1.0.3  Enabled  Crit Powered    6720 AccessPoint 2     7000 [C]
port1.0.4  Disabled Low  Disabled      0 n/a         n/a   n/a
```

Table 25-3: Parameters in the show power-inline interface command output

Parameter	Description
`Interface`	The PoE port(s) in the format `portx.y.z`, where `x` is the device number, `y` is the module number within the device, and `z` is the PoE port number within the module.
`Admin`	The administrative state of PoE on a PoE port, either **Enabled** or **Disabled**.
`Pri`	The current PoE priorities for PoE ports on the PSE, as configured from a **power-inline priority** command: ■ **Low** displays when the `low` parameter is issued. The lowest priority for a PoE enabled port (default). ■ **High** displays when the `high` parameter is issued. The second highest priority for a PoE enabled port. ■ **Crit** displays when the `critical` parameter is issued. The highest priority for a PoE enabled port.
`Oper`	The current PSE PoE port state when this command was issued: ■ **Powered** displays when there is a PD connected and power is being supplied from the PSE. ■ **Denied** displays when supplying power would make the PSE go over the power budget. ■ **Disabled** displays when the PoE port is administratively disabled. ■ **Off** displays when PoE has been disabled for the port. ■ **Fault** displays when a PSE goes over its power allocation.
`Power`	The power consumption in milliwatts (mW) for the PoE port when this command was entered.
`Device`	The description of the connected PD device if a description has been added with the **power-inline description** command. No description is shown for PDs not configured with the **power-inline description** command.
`Class`	The class of the connected PD, if power is being supplied to the PD from the PSE. See **"Power Classes" on page 24.6** in **Chapter 24, Power over Ethernet Introduction** for further information about PD classes and the power assigned per class.
`Max (mW)`	The power in milliwatts (mW) allocated for the PoE port. Additionally, note the following as displayed per PoE port: ■ **[U]** if the power limit for a port was user configured (with the **power-inline max** command). ■ **[L]** if the power limit for a port was supplied by LLDP. ■ **[C]** if the power limit for a port was supplied by the PD class.

Related Commands

show power-inline
show power-inline interface detail

show power-inline interface detail

This command displays detailed information for specified Power over Ethernet (PoE) port(s) on the Power Sourcing Equipment (PSE).

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show power-inline interface [<port-list>] detail`

Parameter	Description
<port-list>	Enter the PoE port(s) to display the PoE port specific information.

Mode User Exec and Privileged Exec

Usage To show detailed PoE information for all ports on the PSE, do not specify any ports.

The power allocated to each port is listed in the `Power allocated` row, and is limited by the maximum power per Powered Device (PD) class, or a user configured power limit.

Example To display detailed PoE port specific information for the port range `1.0.1` to `1.0.2`, use the following command:

```
awplus# show power-inline interface port1.0.1-port1.0.2
       detail
```

Output **Figure 25-5: Example output from the show power-inline interface detail command**

```
awplus#show power-inline interface port1.0.1-1.0.2 detail
Interface port1.0.1
  Powered device type: Desk Phone #1
  PoE admin enabled
  Priority Low
  Detection status: Powered
  Current power consumption: 4800 mW
  Powered device class: 1
  Power allocated: 5000 mW (from configuration)
  Detection of legacy devices is disabled
  Powered pairs: Data
Interface port1.0.2
  Powered device type: Access Point #3
  PoE admin enabled
  Priority High
  Detection status: Powered
  Current power consumption: 6720 mW
  Powered device class: 2
  Power allocated: 7000 mW (from powered device class)
  Detection of legacy devices is enabled
  Powered pairs: Data
```

Table 25-4: Parameters in show power-inline interface detail command output

Parameter	Description
`Interface`	The PoE port(s) in the format `portx.y.z`, where `x` is the device number, `y` is the module number within the device, and `z` is the PoE port number within the module.
`Powered device type:`	The name of the PD, if connected and if power is being supplied to the PD from the PSE, configured with the **power-inline description** command. **n/a** displays if a description has not been configured for the PD.
`PoE admin`	The administrative state of PoE on a PoE capable port, either **Enabled** or **Disabled** as configured from the **power-inline enable** command or the **no power-inline enable** command respectively.
`Priority`	The PoE priority of a port, which is either **Low,** or **High,** or **Critical**, as configured by the **power-inline priority** command.
`Detection status:`	The current PSE PoE port state when this command was issued: ■ **Powered** displays when there is a PD connected and power is being supplied from the PSE. ■ **Denied** displays when supplying power would make the PSE go over the power budget. ■ **Disabled** displays when the PoE port is administratively disabled. ■ **Off** displays when PoE has been disabled for the port. ■ **Fault** displays when a PSE goes over its power allocation.
`Current power consumption:`	The power consumption for the PoE port when this command was entered. Note that the power consumption may have changed since the command was entered and the power is displayed.
`Powered device class:`	The class of the connected PD if connected, and if power is being supplied to the PD from the PSE. See **Chapter 24, Power over Ethernet Introduction** chapter for further information about PD classes and the power assigned per class.
`Power allocated:`	The power in milliwatts (mW) allocated for the PoE port. Additionally, note the following as displayed per PoE port: ■ **[U]** if the power limit for a port was user configured (with the **power-inline max** command). ■ **[L]** if the power limit for a port was supplied by LLDP. ■ **[C]** if the power limit for a port was supplied by the PD class.
`Detection of legacy devices is`	`[Enabled|Disabled]` The status of legacy PoE detection on the PoE port, as configured for the PoE port with the **power-inline allow-legacy** command.
`Powered pairs:`	`[Data|Spare]` The IEEE 802.3af and IEEE 802.3at standards allow for either **data** or **spare** twisted pairs to be used to transfer power to a PD. The powered pairs status for each port. AlliedWare Plus™ PoE switches implement IEEE 802.3af and IEEE 802.3at Endpoint PSE Alternative A (**Data**).

Related Commands

show power-inline
show power-inline interface

Chapter 26: GVRP Introduction and Configuration

Introduction 26.2
GVRP Example 26.3
GVRP Guidelines 26.4
GVRP and Network Security 26.5
GVRP-Inactive Intermediate Switches 26.5
Enabling GVRP on the Switch 26.5
Enabling GVRP on the Ports 26.6
Setting the GVRP Timers 26.6
Disabling GVRP on the Ports 26.7
Disabling GVRP on the Switch 26.7
Configuring and Validating GVRP 26.8

Introduction

GVRP enables the automatic VLAN configuration of switches in a network by allowing GVRP enabled switches to dynamically exchange VLAN configuration information with each other.

GVRP is based on GARP, which defines how attributes, like VIDs, are registered and deregistered. This makes it easier to manage VLANs that span more than one switch. Without GVRP, you have to manually configure your switches to ensure that the various parts of the VLANs can communicate with each other across the different switches. With GVRP this is done for you automatically.

The switch uses GVRP protocol data units (PDUs) to share VLAN information among GVRP-active devices. The PDUs contain the VID numbers of all the VLANs on the switch. When the switch receives a GVRP PDU on a port, it examines the PDU to determine the VIDs of the VLANs on the device that sent it. It then does the following:

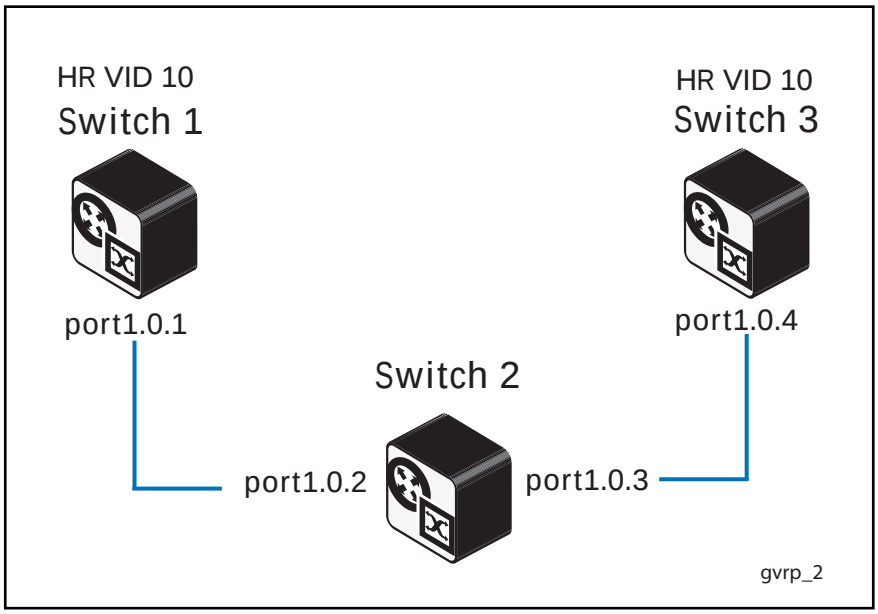
- If the PDU contains a VID of a VLAN that does not exist on the switch, it creates this VLAN and adds the port that received the PDU as a tagged member of the VLAN. A VLAN created by GVRP is called a dynamic GVRP VLAN.
- If the PDU contains a VID of a VLAN that already exists on the switch but the receiving port is not a member of it, the switch adds the port as a tagged member of the VLAN. A port that has been added by GVRP to a static VLAN (that is a user-created VLAN) is called a dynamic GVRP port.

Only GVRP can modify or delete dynamic GVRP VLANs. Dynamic GVRP VLANs exist only so long as the switch continues to receive GVRP PDUs that contain the VID of that VLAN. If there are no more relevant GVRP PDUs arriving, or there are no active links in the VLAN, GVRP deletes it from the switch.

A dynamic GVRP port in a static VLAN remains a member of the VLAN as long as the switch continues to receive GVRP PDUs that contain the VID of that VLAN. If the relevant GVRP PDUs are no longer being received on the port, then GVRP removes the dynamic port from the VLAN, but does not delete the VLAN if the VLAN is a static VLAN.

GVRP Example

The example consists of three switches. Switch 1 and Switch 3 have the HR VLAN 10, but Switch 2 is not configured with the HR VLAN 10. Consequently, the end nodes of the two parts of the HR VLAN 10 cannot communicate with each other because Switch 2 does not have VLAN 10.



Without GVRP, you would have to manually add the HR VLAN 10 to Switch 2. But with GVRP, the VLAN is added automatically. Here is how GVRP resolves this example.

1. Interface `port1.0.1` on Switch 1 sends a PDU (Protocol Data Unit) to interface `port1.0.2` on Switch 2 that contains the VIDs of all the VLANs on Switch 1, including VID 10 for the HR VLAN.

2. Switch 2 examines the PDU it receives on interface `port1.0.2` and finds that it does not have a VLAN with a VID 10. In response, it creates the VLAN as a dynamic GVRP VLAN, assigning it VID 10. Switch 2 then adds interface `port1.0.2`, the switch port that received the PDU, as a tagged member of HR VLAN 10.

3. Switch 2 sends a PDU from interface `port1.0.3` containing all the VIDs of the VLANs on the switch, including the new VID 10. Note at this point interface `port1.0.3` is not a member of VLAN 10. Ports are added to VLANs when they receive PDUs from other switches in the network, not when they transmit PDUs.

4. Switch 3 receives the PDU on interface port1.0.4 and, after examining it, finds that one of the VLANs on Switch 2 has the VID 10, which matches the VID of an already existing VLAN on the switch. So it does not create the VLAN because it already exists. It then determines whether the port that received the PDU, in this case interface `port1.0.4`, is a member of the VLAN. If it is not a member, it adds the port to the VLAN as a tagged dynamic GVRP port. If the port is already a member of the VLAN, then no change is made.

5. Switch 3 sends a PDU out interface `port1.0.4` to interface `port1.0.3` on Switch 2.

6. Switch 2 receives the PDU on interface `port1.0.3` and then adds the port as a tagged dynamic GVRP port to the dynamic GVRP VLAN 10.

There is now a communications path for the end nodes of the HR VLAN 10 on Switch 1 and Switch 3. GVRP created the new dynamic GVRP VLAN with a VID of 10 on Switch 2 and added interfaces `port1.0.2` and `port1.0.3` to HR VLAN 10 as tagged dynamic GVRP ports.

GVRP Guidelines

Here are the guidelines for configuring GVRP on your switch:

- All ports that constitute a network link between the switch and the other switches must be running GVRP.
- You cannot modify or delete dynamic GVRP VLANs.
- You cannot remove dynamic GVRP ports from static or dynamic VLANs.
- There is limit of 400 VLANs supported by the AlliedWare Plus GVRP implementation. VLANs may be numbered 1-4094, but a limit of 400 of these VLANs are supported.
- MSTP is not supported by the current AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.
- VCStack is not supported by the current AlliedWare Plus GVRP implementation.
- To be detected by GVRP, a VLAN must have at least one active port. GVRP cannot detect a VLAN that does not have any active nodes or valid port links.
- Rebooting the switch erases all dynamic GVRP VLANs and dynamic GVRP port assignments. The dynamic assignments are relearned by the switch as PDUs arrive on the ports from other switches.
- GVRP has three timers: join timer, leave timer, and leave all timer. The values for these timers must be set the same on all switches running GVRP. Timers with different values on different switches can result in GVRP compatibility problems.
- You can convert dynamic GVRP VLANs and dynamic GVRP port assignments to static VLANs and static port assignments.
- The default port settings on the switch for GVRP is inactive, meaning that the ports will not participate in GVRP until enabled on the switch globally and on the interface locally.
- Allied Telesis recommends disabling GVRP on those ports that are connected to GVRP-inactive devices, meaning any switches that do not have the GVRP feature enabled.
- PDUs are transmitted from only those switch ports where GVRP is enabled.
- Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. GVRP and private VLAN trunk ports are mutually exclusive.

GVRP and Network Security

GVRP should be used with caution because it can expose your network to unauthorized access. If a network intruder were to connect to a switch port running GVRP and transmit a bogus GVRP PDU containing VIDs of restricted VLANs, GVRP would make the port a member of the VLANs, giving the intruder access to restricted areas of your network.

Here are a few suggestions to protect against this type of unauthorized network intrusion:

- Activating GVRP only on those switch ports connected to other GVRP devices. Do not activate GVRP on ports that are connected to GVRP inactive devices.
- Converting all dynamic GVRP VLANs and dynamic GVRP ports to static assignments, and then turning off GVRP on all the switches. This preserves the new VLAN assignments while protecting against unauthorized network intrusion.

GVRP-Inactive Intermediate Switches

If two GVRP-active devices are separated by a GVRP-inactive switch, the GVRP-active devices may not be able to share VLAN information. There are two issues involved.

The first is whether the intermediate switch forwards the GVRP PDUs that it receives from the GVRP-active switches. GVRP PDUs are management frames, intended for the switch's CPU. In all likelihood, a GVRP-inactive switch will discard the PDUs because it will not recognize them.

The second issue is that even if a GVRP-inactive switch forwards GVRP PDUs, it will not automatically create the VLANs. Consequently, even if GVRP-active switches receive the PDUs and create the necessary VLANs, an intermediate switch may block the VLAN traffic, unless you modify its VLANs and port assignments manually.

Enabling GVRP on the Switch

The command for enabling GVRP on the switch is found in the Global Configuration mode. It is the **gvrp enable (global)** command. After the command is entered, the switch immediately begins to transmit PDUs from those ports where GVRP is enabled.

Further, to enable the switch to create dynamic VLANs if it receives GVRP PDUs that contain VIDs for VLANs it does not currently have, use the command **gvrp dynamic-vlan-creation**.

Here are the commands to enable GVRP on the switch and enable to switch to create dynamic VLANs if it receives GVRP PDUs that contain VIDs for VLANs it does not currently have:

```
awplus>enable
awplus#configure terminal
awplus(config)#gvrp enable
awplus(config)#gvrp dynamic-vlan-creation
```

For reference information, refer to the **gvrp enable (global)** command and the **gvrp dynamic-vlan-creation** command in the **GVRP Commands** chapter.

Enabling GVRP on the Ports

To activate GVRP on the ports so that they transmit GVRP PDUs, use the **gvrp registration** and the **gvrp (interface)** commands in the Interface Configuration mode. Because the default setting for GVRP on the ports is disabled, you need to use these commands if you want to re-enable GVRP after disabling it on a port.

This example of these commands activates GVRP on interface `port1.0.4`,`port1.0.5`, and `port1.0.6`:

```
awplus>enable
awplus#configure terminal
awplus(config)#interface port1.0.4,port1.0.5,port1.0.6
awplus(config-if)#gvrp registration normal
awplus(config-if)#gvrp
```

For reference information, refer to the **gvrp registration** and **gvrp (interface)** commands in the **GVRP Commands** chapter.

Setting the GVRP Timers

The switch has a join timer, a leave timer, and a leave all timer. You should not change the timers unless you understand their functions. (Refer to the IEEE 802.1p standard for the timer definitions.) The timers have to be set the same on all GARP-active network devices and the join timer and the leave timer have to be set according to the following rule:

leave timer >= (3 x (join timer))

When configuring the leave timer, set it to more than or equal to three times the join timer value. The settings for the leave and join timers must be the same for all GVRP enabled switches.

The commands for setting the timers are in the Interface Configuration mode. They are:

gvrp timer join
gvrp timer leave
gvrp timer leaveall

The timers are set in one hundredths of a second. This example sets the join timer to 0.2 seconds, the leave timer to 0.8 seconds and the leave all timer to 10 seconds for port1.0.2:

```
awplus>enable
awplus#configure terminal
awplus(config)#interface port1.0.2
awplus(config-if)#gvrp timer join 20
awplus(config-if)#gvrp timer leave 80
awplus(config-if)#gvrp timer leaveall 1000
```

For reference information, refer to **gvrp timer** command in the **GVRP Commands** chapter.

Disabling GVRP on the Ports

To disable GVRP on the ports, use the **gvrp registration none** and **no gvrp (interface)** commands in the Interface Configuration mode.

This example of the command deactivates GVRP on interfaces `port1.0.4` and `port1.0.5`:

```
awplus>enable
awplus#configure terminal
awplus(config)#interface port1.0.4,port1.0.5
awplus(config-if)#gvrp registration none
awplus(config-if)#no gvrp
```

For reference information, refer to **gvrp registration** and **gvrp (interface)** command in the **GVRP Commands** chapter.

Disabling GVRP on the Switch

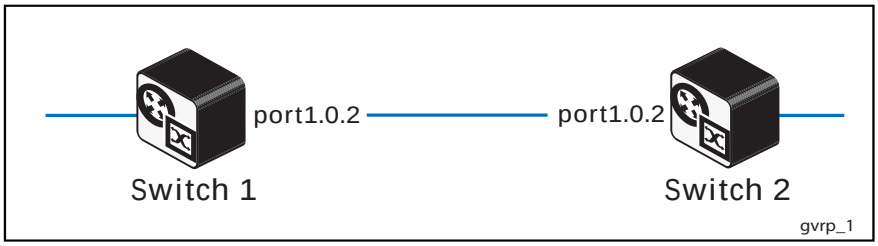
To disable GVRP to stop the switch from learning any further dynamic VLANs or GVRP ports, use the **no gvrp (interface) enable** command in the Global Configuration mode. Here is the command.

```
awplus>enable
awplus#configure terminal
awplus(config)#no gvrp enable
```

For reference information, refer to the **gvrp (interface)** command in the **GVRP Commands** chapter.

Configuring and Validating GVRP

GVRP (GARP VLAN Registration Protocol) allows the exchange of VLAN information between switches in a network. If one switch is manually configured with multiple VLANs, other switches in the network learn about these VLANs dynamically through GVRP.



Switch 1: Configuring GVRP to receive VLANs from Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode from the Privileged Exec mode.
`awplus(config)#` `gvrp enable`	Enter GVRP on **Switch 1**.
`awplus(config)#` `gvrp dynamic-vlan-creation`	Enable dynamic VLAN creation for GVRP. Note that GVRP is now enabled globally for **Switch 1**.
`awplus(config)#` `interface port1.0.2`	Specify an interface (`port1.0.2`) to be configured and enter Interface Configuration mode.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of the interface as trunk and specify tagged frames only. Any frames not tagged as trunk frames are discarded.
`awplus(config-if)#` `switchport trunk allowed vlan all`	Apply to all VLANs on this interface.
`awplus(config-if)#` `gvrp`	Enable GVRP on switch port `port1.0.2`. Note that GVRP is now set up on interface `port1.0.2` as GVRP is also enabled globally for **Switch 1**.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `exit`	Exit Global Configuration mode and enter Privileged Exec mode.
`awplus#` `show gvrp configuration`	Show GVRP configuration on **Switch 1** to confirm GVRP is ready to propagate VLANs.

Switch 2: Configuring GVRP & creating VLANs to propagate:

`awplus#` `enable`	Enter the Privileged Exec mode.
`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `gvrp enable`	Enter GVRP on **Switch 2**.
`awplus(config)#` `vlan database`	Create VLANs to propagate between **Switch 1** and **Switch 2** with GVRP enabled on the Switches and on the interfaces on each Switch.
`awplus(config-vlan)#` `vlan 20-30`	Create 11 VLANs with VIDs 20 through 30 to propagate between interface `port1.0.2` on **Switch 1** and **Switch 2**.
`awplus(config)#` `gvrp dynamic-vlan-creation`	Enable dynamic VLAN creation for GVRP. Note that GVRP is now enabled globally for **Switch 2**.
`awplus(config)#` `interface port1.0.2`	Specify an interface (`port1.0.2`) to be configured and enter Interface Configuration mode.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of the interface as trunk and specify tagged frames only. Any frames not tagged as trunk frames are discarded.
`awplus(config-if)#` `switchport trunk allowed vlan all`	Set this interface to be a tagged member of all VLANs.
`awplus(config-if)#` `gvrp`	Enable GVRP on switch port `port1.0.2`.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `exit`	Exit Global Configuration mode and enter Privileged Exec mode.
`awplus#` `show gvrp configuration`	Show GVRP configuration on **Switch 2** to confirm GVRP is ready to propagate VLANs.

Switch 1: Validating VLANs have propagated from Switch 2:

`awplus#`	
`show vlan`	Confirm the VLANs are available from **Switch 2** on **Switch 1** by examining show output to confirm VLANs from **Switch 2** are on **Switch 1**.

Names of commands used

gvrp (interface)
gvrp dynamic-vlan-creation
switchport mode trunk
vlan database
vlan

Validation commands

show vlan

Chapter 27: GVRP Commands

Command List 27.2
clear gvrp statistics 27.2
debug gvrp 27.3
gvrp (interface) 27.4
gvrp dynamic-vlan-creation 27.5
gvrp enable (global) 27.6
gvrp registration 27.7
gvrp timer 27.8
show debugging gvrp 27.10
show gvrp configuration 27.11
show gvrp machine 27.12
show gvrp statistics 27.13
show gvrp timer 27.14

Command List

With GVRP enabled the switch can exchange VLAN configuration information with other GVRP enabled switches. VLANs can be dynamically created and managed through trunk ports.

- There is limit of 400 VLANs supported by the AlliedWare Plus GVRP implementation. VLANs may be numbered 1-4094, but a limit of 400 of these VLANs are supported.
- MSTP is not supported by the AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.
- VCStack is not supported by the current AlliedWare Plus GVRP implementation.

This chapter provides an alphabetical reference for commands used to configure GVRP. For information about GVRP, including configuration, see **Chapter 26, GVRP Introduction and Configuration**.

clear gvrp statistics

Use this command to clear the GVRP statistics for all switchports, or for a specific switchport.

Syntax `clear gvrp statistics {all|<interface>}`

Parameter	Description
`all`	Specify all switchports to clear GVRP statistics.
`<interface>`	Specify the switchport to clear GVRP statistics.

Mode Privileged Exec

Usage Use this command together with the **show gvrp statistics** command to troubleshoot GVRP.

Examples To clear all GVRP statistics for all switchport on the switch, enter the command:

```
awplus#clear gvrp statistics all
```

To clear GVRP statistics for switchport interface `port1.0.3`, enter the command:

```
awplus#clear gvrp statistics port1.0.3
```

Related Commands **show gvrp statistics**

debug gvrp

Use this command to debug GVRP packets and commands, sending output to the console.

Use the **no** variant of this command to turn off debugging for GVRP packets and commands.

Syntax

```
debug gvrp {all|cli|event|packet}
no debug gvrp {all|cli|event|packet}
```

Parameter	Description
`all`	Specifies debugging for all levels.
`cli`	Specifies debugging for commands.
`event`	Specified debugging for events.
`packet`	Specifies debugging for packets.

Mode Privileged Exec and Global Configuration

Examples To send debug output to the console for GVRP packets and GVRP commands, and to enable the display of debug output on the console first, enter the commands:

```
awplus#terminal monitor
awplus#configure terminal
awplus(config)#debug gvrp all
```

To send debug output for GVRP packets to the console, enter the commands:

```
awplus#terminal monitor
awplus#configure terminal
awplus(config)#debug gvrp packets
```

To send debug output for GVRP commands to the console, enter the commands:

```
awplus#terminal monitor
awplus#configure terminal
awplus(config)#debug gvrp cli
```

To stop sending debug output for GVRP packets and GVRP commands to the console, and to stop the display of any debug output on the console, enter the commands:

```
awplus#terminal no monitor
awplus#configure terminal
awplus(config)#no debug gvrp all
```

Related Commands **show debugging gvrp**
terminal monitor

gvrp (interface)

Use this command to enable GVRP for switchport interfaces.

Use the **no** variant of this command to disable GVRP for switchport interfaces.

Syntax

```
gvrp
no gvrp
```

Mode Interface Configuration (for switchport interfaces).

Default Disabled by default.

Usage Use this command to enable GVRP on switchport interfaces. Note this command does not enable GVRP for the switch. To enable GVRP on switchports use this command in Interface Configuration mode. You must issue a **gvrp enable (global)** command before issuing a **gvrp (interface)** command.

You must enable GVRP on both ends of a link for GVRP to propagate VLANs between links.

Note

MSTP is not supported by the current AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.

Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. GVRP and private VLAN trunk ports are mutually exclusive.

Examples To enable GVRP on interfaces `port1.0.1-port1.0.2`, enter the commands:

```
awplus#configure terminal
awplus(config)#gvrp enable
awplus(config)#interface port1.0.1-port1.0.2
awplus(config-if)#gvrp
```

To disable GVRP on interfaces `port1.0.1-port1.0.2`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1-port1.0.2
awplus(config-if)#no gvrp
```

Validation Commands **show gvrp configuration**

Related Commands **gvrp dynamic-vlan-creation**
gvrp enable (global)

gvrp dynamic-vlan-creation

Use this command to enable dynamic VLAN creation globally for the switch.

Use the **no** variant of this command to disable dynamic VLAN creation globally for the switch.

Syntax

```
gvrp dynamic-vlan-creation
no gvrp dynamic-vlan-creation
```

Mode Global Configuration

Default Disabled by default.

Usage You must enable GVRP on both ends of a link for GVRP to propagate VLANs between links.

You must also enable GVRP globally in Global Configuration mode before enabling GVRP on an interface in Interface Configuration mode. Both of these tasks must occur to create VLANs.

Note There is limit of 400 VLANs supported by the AlliedWare Plus GVRP implementation. VLANs may be numbered 1-4094, but a limit of 400 of these VLANs are supported.

Examples To enable GVRP dynamic VLAN creation on the switch, enter the commands:

```
awplus#configure terminal
awplus(config)#gvrp enable
awplus(config)#gvrp dynamic-vlan-creation
```

Enter the following commands for switches with hostnames `awplus_switch1` and `awplus_switch2` respectively, so `awplus_switch1` propagates VLANs to `awplus_switch2` and `awplus_switch2` propagates VLANs to `awplus_switch1`:

```
awplus_switch1#configure terminal
awplus_switch1(config)#gvrp enable
awplus_switch1(config)#gvrp dynamic-vlan-creation

awplus_switch2#configure terminal
awplus_switch2(config)#gvrp enable
awplus_switch2(config)#gvrp dynamic-vlan-creation
```

To disable GVRP dynamic VLAN creation on the switch, enter the commands:

```
awplus#configure terminal
awplus(config)#no gvrp dynamic-vlan-creation
```

Validation Commands **show gvrp configuration**

Related Commands **gvrp enable (global)**

gvrp enable (global)

Use this command to enable GVRP globally for the switch.

Use the **no** variant of this command to disable GVRP globally for the switch.

Syntax

```
gvrp enable
no gvrp enable
```

Mode Global Configuration

Default Disabled by default.

Usage Use this command to enable GVRP on the switch. Note that this command does not enable GVRP on switchports. To enable GVRP on switchports use the **gvrp (interface)** command in Interface Configuration mode. You must issue a **gvrp enable (global)** command before issuing a **gvrp (interface)** command.

You must enable GVRP on both ends of a link for GVRP to propagate VLANs between links.

Note

MSTP is not supported by the current AlliedWare Plus GVRP implementation. GVRP and MSTP are mutually exclusive. STP and RSTP are supported by GVRP.

Private VLAN trunk ports are not supported by the current AlliedWare Plus GVRP implementation. GVRP and private VLAN trunk ports are mutually exclusive.

Examples To enable GVRP for the switch, before enabling GVRP on switchports, enter the commands:

```
awplus#configure terminal
awplus(config)#gvrp enable
```

To disable GVRP on the switch, which will also disable GVRP enabled on switchports, enter the commands:

```
awplus#configure terminal
awplus(config)#no gvrp enable
```

Validation Commands **show gvrp configuration**

Related Commands **gvrp (interface)**
gvrp dynamic-vlan-creation

gvrp registration

Use this command to set GVRP registration to normal, fixed, and forbidden registration modes.

Use the **no** variant of this command to disable GVRP registration.

Syntax

```
gvrp registration {normal|fixed|forbidden}
no gvrp registration {normal|fixed|forbidden}
```

Parameter	Description
`normal`	Specify dynamic GVRP registration and deregistration of VLANs.
`fixed`	Specify fixed GVRP registration and deregistration of VLANs.
`forbidden`	Specify no GVRP registration of VLANs. VLANs are deregistered.

Mode Interface Configuration

Default Normal registration is the default.

Usage Configuring a trunk port in normal registration mode allows dynamic creation of VLANs. Normal mode is the default mode. Validate using the **show gvrp configuration** command.

Configuring a trunk port in fixed registration mode allows manual creation of VLANs.

Configuring a trunk port in forbidden registration mode prevents VLAN creation on the port.

Examples To configure GVRP registration to `fixed` on `port1.0.1`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1
awplus(config-if)#gvrp registration fixed
```

To disable GVRP on interfaces `port1.0.1-port1.0.2`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1-port1.0.2
awplus(config-if)#no gvrp
```

Validation Commands **show gvrp configuration**

gvrp timer

Use this command to set GVRP timers in Interface Configuration mode for a given interface.

Use the **no** variant of this command to reset the GVRP timers to the defaults specified in the table below.

Syntax

```
gvrp timer {join <timer-value>|leave <timer-value>|
    leaveall <timer-value>}

no gvrp timer {join|leave|leaveall}
```

Parameter	Description
`join`	Specifies the timer for joining the group (default is 20 centiseconds / hundredths of a second, or 200 milliseconds).
`leave`	Specifies the timer for leaving a group (default is 60 centiseconds / hundredths of a second, or 600 milliseconds).
`leaveall`	Specifies the timer for leaving all groups (default is 1000 centiseconds / hundredths of a second, or 10,000 milliseconds).
<timer-value>	<1-65535> The timer value in hundredths of a second (centiseconds).

Mode Interface Configuration

Defaults The default join time value is 20 centiseconds (200 milliseconds), the default leave timer value is 60 centiseconds (600 milliseconds), and the default leaveall timer value is 1000 centiseconds (10,000 milliseconds).

Usage When configuring the `leave` timer, set it to more than or equal to three times the `join` timer value. The settings for the `leave` and `join` timers must be the same for all GVRP enabled switches. See also the section **"Setting the GVRP Timers" on page 26.6**.

Use the **show gvrp timer** command to confirm GVRP timers set with this command.

Examples To set the GVRP `join` timer to 300 hundredths of a second for interface `port1.0.1`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1
awplus(config-if)#gvrp timer join 20
```

To set the GVRP `leave` timer to 600 hundredths of a second for interface `port1.0.2`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.2
awplus(config-if)#gvrp timer leave 60
```

To set the GVRP `leaveall` timer to 1000 hundredths of a second for interface `port1.0.1`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1
awplus(config-if)#gvrp timer leaveall 1000
```

To reset the GVRP `join` timer to its default (200 milliseconds) for interface `port1.0.1`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1
awplus(config-if)#no gvrp timer join
```

To reset the GVRP `leave` timer to its default (600 milliseconds) for interface `port1.0.2`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.2
awplus(config-if)#no gvrp timer leave
```

To disable GVRP on interfaces `port1.0.1-port1.0.2`, enter the commands:

```
awplus#configure terminal
awplus(config)#interface port1.0.1-port1.0.2
awplus(config-if)#no gvrp
```

Related Commands **show gvrp timer**

show debugging gvrp

Use this command to display the GVRP debugging option set.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging gvrp`

Mode User Exec and Privileged Exec

Example Enter the following commands to display GVRP debugging output on the console:

```
        awplus#configure terminal
awplus(config)#debug gvrp all
awplus(config)#exit
        awplus#show debugging gvrp
```

Output See sample output from the `show debugging gvrp` after entering `debug gvrp all`:

```
GVRP debugging status:
  GVRP Event debugging is on
  GVRP CLI debugging is on
  GVRP Timer debugging is on
  GVRP Packet debugging is on
```

Related Commands **debug gvrp**

show gvrp configuration

Use this command to display GVRP configuration data for a switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show gvrp configuration`

Mode User Exec and Privileged Exec

Example To show GVRP configuration for the switch, enter the command:

```
awplus#show gvrp configuration
```

Output The following is an output of this command displaying the GVRP configuration for a switch:

```
awplus#show gvrp configuration
Global GVRP Configuration:
GVRP Feature: Enabled
Dynamic Vlan Creation: Disabled
Port based GVRP Configuration:
                                                  Timers(centiseconds)
Port    GVRP Status Registration  Applicant   Join   Leave
LeaveAll
-----------------------------------------------------------------
port1.0.1 Enabled   Normal        Normal      20       60    1000
port1.0.2 Enabled   Normal        Normal      200     600   10000
```

show gvrp machine

Use this command to display the state machine for GVRP.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show gvrp machine`

Mode User Exec and Privileged Exec

Example To show the GVRP state machine for the switch, enter the command:

```
awplus#show gvrp machine
```

Output See the following output of this command displaying the GVRP state machine.

```
awplus show gvrp machine
  port = 1.0.1  applicant state = QA   registrar state = INN
  port = 1.0.2  applicant state = QA   registrar state = INN
```

show gvrp statistics

Use this command to display a statistical summary of GVRP information for the switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show gvrp statistics [<interface>]`

Parameter	Description
<interface>	The name of the switchport interface.

Mode User Exec and Privileged Exec

Usage Use this command together with the **clear gvrp statistics** command to troubleshoot GVRP.

Examples To show the GVRP statistics for all switchport interfaces, enter the command:

```
awplus#show gvrp statistics
```

To show the GVRP statistics for switchport interfaces `port1.0.1` and `port1.0.2`, enter the command:

```
awplus#show gvrp statistics port1.0.1-port1.0.2
```

Output The following is an output of this command displaying a statistical summary for `port1.0.1-port1.0.2`

```
awplus# show gvrp statistics port1.0.1-port1.0.2
Port     JoinEmpty    JoinIn   LeaveEmpty     LeaveIn     Empty
-----------------------------------------------------------------
1.0.1    RX           0          2          0         0       0
         TX           0          0          0         0       0
1.0.2    RX           0          1          0         0       1
         TX           0          0          0         0       0
```

Related Commands **clear gvrp statistics**

show gvrp timer

Use this command to display data for the GVRP timers set with the **gvrp timer** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show gvrp timer <interface>`

Parameter	Description
<interface>	The name of the switchport interface.

Mode User Exec and Privileged Exec

Examples To show the GVRP timers for all switchport interfaces, enter the command:

```
awplus#show gvrp timer
```

To show the GVRP timers for switchport interface `port1.0.1`, enter the command:

```
awplus#show gvrp timer port1.0.1
```

Output The following show output displays data for timers on the switchport interface `port1.0.1`

```
awplus# show gvrp timer port1.0.1
Timer           Timer Value (centiseconds)
----------------------------------------
Join                  20
Leave                 60
Leave All             1000
```

Related Commands **gvrp timer**

Part 3: Layer Three Switching and Routing

- Chapter 28 Internet Protocol (IP) Addressing and Protocols
- Chapter 29 IP Addressing and Protocol Commands
- Chapter 30 IPv6 Introduction
- Chapter 31 IPv6 Commands
- Chapter 32 IPv6to4 Tunneling Configuration
- Chapter 33 IPv6to4 Tunneling Commands
- Chapter 34 Routing Protocol Overview
- Chapter 35 Route Selection
- Chapter 36 Routing Commands
- Chapter 37 RIP Configuration
- Chapter 38 RIP Commands
- Chapter 39 RIPng for IPv6 Configuration
- Chapter 40 RIPng for IPv6 Commands
- Chapter 41 OSPF Introduction and Configuration
- Chapter 42 OSPF Commands
- Chapter 43 OSPFv3 for IPv6 Introduction and Configuration
- Chapter 44 OSPFv3 for IPv6 Commands
- Chapter 45 Route Map Configuration
- Chapter 46 Route Map Commands

Chapter 28: Internet Protocol (IP) Addressing and Protocols

Introduction 28.2
Address Resolution Protocol (ARP) 28.3
- Static ARP Entries 28.3
- Timing Out ARP Entries 28.3
- Deleting ARP Entries 28.4
- Proxy ARP 28.4
- ARP Logging 28.7

Domain Name System (DNS) 28.8
- Domain Name Parts 28.8
- Server Hierarchy 28.8
- DNS Client 28.9
- DNS Relay 28.11
- DHCP Options 28.12

Internet Control Message Protocol (ICMP) 28.13
Checking IP Connections 28.14
- Ping 28.14
- Traceroute 28.14

IP Helper 28.15
IP Directed Broadcast 28.16

Introduction

This chapter describes how to configure IPv4 addressing and the protocols used to help IP function on your network.

As well as the familiar Internet (with uppercase "I"), the term internet (with lowercase "i") can refer to any network (usually a wide area network) that uses the Internet Protocol. This chapter concentrates on this definition—a generalized network that uses IP as its network protocol.

Assigning an IP Address

To configure your device to perform IP routing (for example, to access the Internet) you need to configure IP. You also need to configure IP if you want to manage your device from any IP-based management process (such as SSH, Telnet, or SNMP).

Add an IP address to each of the interfaces that you want to process IP traffic.

You can configure an interface on your device with a static IP address, or with a dynamic IP address assigned using your device's DHCP client.

Static IP addresses

To add a static IP address to an interface, enter interface mode for the interface that you want to configure, then use the command:

```
awplus(config-if)# ip address <ip-addr/prefix-length>
                   [secondary] [label <label>]
```

where `<ip-address/m>` the IP address followed by a slash then the prefix length. Note that you cannot specify the mask in dotted decimal notation in this command.

For example, to give the interface vlan1 an address of 192.168.10.10, with a class C subnet mask, use the command:

```
awplus(config-if)# ip address 192.168.10.10/24
```

The **secondary** parameter allows you to add multiple IP addresses to an interface using this command. Each interface must have a primary IP address before you can add a secondary address. Your device treats secondary addresses the same as primary addresses in most instances, such as responding for ARP requests for the IP address. However, the only packets generated that have a secondary address as source address are routing updates. You can define up to 32 secondary addresses on a single interface.

DHCP dynamic addresses

When you use the DHCP client, it obtains the IP address and subnet mask for the interface, and other IP configuration parameters, from a DHCP server. To configure an interface to gain its IP configuration using the DHCP client, use the command:

```
awplus(config-if)# ip address dhcp [client-id <interface>]
                   [hostname <hostname>]
```

If an IP interface is configured to get its IP address and subnet mask from DHCP, the interface does not take part in IP routing until the IP address and subnet mask have been set by DHCP.

If you need to make a static entry in your DHCP server for the device, you need your device's MAC address, which you can display by using the command:

```
awplus# show interface
```

See **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction** for more information about DHCP.

Address Resolution Protocol (ARP)

Address Resolution Protocol (ARP) is used by your device to dynamically learn the Layer 2 address of devices in its networks. Most hosts also have a MAC physical address in addition to the assigned IP address. For Ethernet, this is a 6-byte, globally unique number. ARP enables your device to learn the physical address of the host that has a given IP address.

When your device needs to forward packets to a destination that it does not know the Layer 2 address of, it broadcasts an ARP request to determine where to send the packet. The ARP request is a broadcast packet and includes the target IP address. All stations on the LAN receive this broadcast but only one host recognizes its own IP address. It replies, thereby giving your device its physical address.

Your device creates a dynamic ARP entry in its ARP cache, to record the IP address to physical address mapping (also called a binding). It uses that ARP entry to forward further packets to that address.

The ARP protocol is described in RFC 826, ***An Ethernet Address Resolution Protocol—or—Converting Network Protocol Addresses to 48 bit Ethernet Address for Transmission on Ethernet Hardware***.

Static ARP Entries

If your LAN includes hosts that do not support ARP, you can add a static ARP entry to the cache. However, it is rarely necessary to add an ARP entry this way. To add a static ARP entry, use the command:

```
awplus(config)# arp <ip-addr> <mac-address> [<port-number>]
                [alias]
```

Timing Out ARP Entries

Your device times out dynamic ARP entries to ensure that the cache does not fill with entries for hosts that are no longer active. If your device stops receiving traffic for a device specified in a dynamic ARP entry, it deletes the ARP entry after a configurable timeout period. Static ARP entries are not aged or automatically deleted.

Increasing the ARP timeout reduces the amount of network traffic. Decreasing the timeout makes your device more responsive to changes in network topology.

To set a timeout period, enter the interface mode, then use the command:

```
awplus(config-if)# arp-aging-timeout <0-432000>
```

Deleting ARP Entries

To remove a static ARP entry, use the command:

```
awplus(config)# no arp <ip-addr>
```

To clear the ARP cache of dynamic entries, use the command:

```
awplus# clear arp-cache
```

This removes the dynamic ARP entries for all interfaces.

To display the entries in the ARP cache, use the command:

```
awplus)# show arp
```

The ARP cache will be repopulated by the normal ARP learning mechanism. As long as the entries are relearned quickly enough, deleting dynamic ARP entries does not affect:

- routes
- OSPF neighbor status
- the TCP/UDP connection status
- VRRP status

Proxy ARP

Proxy ARP (defined in RFC 1027) allows hosts that do not support routing (i.e. they have no knowledge of the network structure) to determine the physical addresses of hosts on other networks. Your device intercepts ARP broadcast packets and substitutes its own physical address for that of the remote host. This occurs only if your device has the best route to the remote host. By responding to the ARP request, your device ensures that subsequent packets from the local host are directed to its physical address, and it can then forward these to the remote host. The process is symmetrical.

Proxy ARP is disabled by default. To enable proxy ARP on an interface, use the commands:

```
awplus(config)# interface <interface>
awplus(config-if)# ip proxy-arp
```

To disable Proxy ARP on an interface, use the command:

```
awplus(config-if)# no ip proxy-arp
```

To check Proxy ARP is enabled on an interface, use the **show running-config** command. If Proxy ARP has been enabled an entry shows **ip proxy-arp** below the interface it is enabled on. No **ip proxy-arp** entry below an interface in the config indicates Proxy ARP is disabled on it.

See the sample configuration commands and validation command with resulting output showing proxy ARP **enabled** on VLAN 2 below:

```
awplus#configure terminal
awplus(config)#interface vlan2
awplus(config-if)#ip proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan2
 ip proxy-arp
 ip address 192.168.2.2/24
!
```

See the sample configuration commands and validation command with resulting output showing proxy ARP **disabled** on VLAN 2 below:

```
awplus#configure terminal
awplus(config)#interface vlan2
awplus(config-if)#no ip proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan2
 ip address 192.168.2.2/24
!
```

Local Proxy ARP

Local Proxy ARP lets you stop MAC address resolution between hosts within an interface's subnet. This ensures that devices within a subnet cannot send traffic that bypasses Layer 3 routing on your device. This lets you monitor, filter, and control traffic between devices in the same subnet.

Local Proxy ARP extends proxy ARP by intercepting and responding to ARP requests between hosts within a subnet. Local proxy ARP responds to ARP requests with your device's own MAC address details instead of those from the destination host. This stops hosts from learning the MAC address of other hosts within its subnet.

When Local Proxy ARP is operating on an interface, your device does not generate or forward any ICMP-Redirect messages on that interface.

Local Proxy ARP is disabled by default. To enable local proxy ARP on an interface, use the commands:

```
awplus(config)# interface <interface>
awplus(config-if)# ip local-proxy-arp
```

To disable local proxy ARP on an interface, use the command:

```
awplus(config-if)# no ip local-proxy-arp
```

To check Local Proxy ARP is enabled on an interface, use the **show running-config** command. If Local Proxy ARP has been enabled an entry shows **ip local-proxy-arp** below the interface it is enabled on. No **ip local-proxy-arp** entry below an interface in the config indicates Local Proxy ARP is disabled on it.

See the sample configuration commands and validation command with resulting output showing local proxy ARP **enabled** on VLAN 1 below:

```
awplus#configure terminal
awplus(config)#interface vlan1
awplus(config-if)#ip local-proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan1
 ip local-proxy-arp
 ip address 192.168.1.2/24
!
```

See the sample configuration commands and validation command with resulting output showing Local Proxy ARP **disabled** on VLAN 1 below:

```
awplus#configure terminal
awplus(config)#interface vlan1
awplus(config-if)#no ip local-proxy-arp
awplus(config-if)#end
awplus(config)#exit
awplus#show running-config
!
interface vlan1
 ip address 192.168.1.2/24
!
```

ARP Logging

You can enable your device to log static and dynamic ARP entries, and you can select either default hexadecimal notation (HHHH.HHHH.HHHH) or standard IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH) for the MAC addresses displayed in the ARP log output.

If this feature is enabled, ARP log messages are stored on the device in RAM. If the device is rebooted the ARP log messages are lost. ARP logging is disabled by default.

To enable ARP logging, use the command:

```
awplus(config)# arp log [mac-address-format ieee]
```

You can specify whether the MAC address is displayed in the default hexadecimal notation HHHH.HHHH.HHHH or in the standard IEEE format HH-HH-HH-HH-HH-HH.

To disable ARP logging, use the command:

```
awplus(config)# no arp log [mac-address-format ieee]
```

To display the ARP log messages, use the command:

```
awplus(config)# show log | include ARP_LOG
```

See the sample ARP log output and descriptions of the fields displayed in the sample ARP log output in the **arp log** command on page 29.6.

Domain Name System (DNS)

The Domain Name System allows you to access remote systems by entering human-readable device host names rather than IP addresses. DNS works by creating a mapping between a domain name, such as “www.alliedtelesis.com”, and its IP address. These mappings are held on DNS servers. DNS translates meaningful domain names into IP addresses for networking equipment to locate and address these devices. The benefits of DNS are that domain names:

- can map to a new IP address if the host’s IP address changes
- are easier to remember than an IP address
- allow organizations to use a domain name hierarchy that is independent of any IP address assignment

Your AlliedWare Plus™ device has a the ability to resolve domain names for internally generated commands (DNS Client) as well as providing the DNS information to connected hosts (via DNS Relay, DHCP Server, or DHCP Relay).

The DNS Client is enabled automatically when at least one DNS server is present on the interface. This client allows you to use domain names instead of IP addresses when using commands on your device from this interface.

The DNS Relay provides the presence of a local virtual DNS server which can service DNS lookup requests sent to it from local hosts. The DHCP Server can be configured to provide domain names information to DHCP clients during the lease process.

Domain Name Parts

Domain names are made up of a hierarchy of two or more name segments. Each segment is separated by a period. The format of domain names is the same as the host portion of a URL (Uniform Resource Locator). The first segment from the left is unique to the host, with each following segment mapping the host in the domain name hierarchy. The segment on the far right is a top-level domain name shared by many hosts.

Server Hierarchy

A network of domain name servers maintains the mappings between domain names and their IP addresses. This network operates in a hierarchy that is similar to the structure of the domain names. When a local DNS server cannot resolve your request it sends the request to a higher level DNS server.

For example, to access the site “alliedtelesis.com”, your PC sends a DNS inquiry to its local DNS server asking for the IP address matching alliedtelesis.com. If this address is already locally cached (following its recent use), the DNS server returns the IP address that matches alliedtelesis.com. If the DNS server does not have this address cached, it forwards the request upwards through the hierarchy of DNS servers until a DNS server can resolve the mapping. This means an often-used domain name is resolved quickly, while an uncommon or nonexistent domain may take longer to resolve or fail.

As well as the hierarchy of domain name servers accessible through the Internet, you can operate your own DNS server to map to private IP addresses within your network.

The DHCP server IP address can be either statically defined, or can be dynamically assigned via DHCPv4 option 6 using **"ip name-server" on page 29.31** and DHCP option 15 using **"ip domain-name" on page 29.24** if DHCP client is configured. See **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction** for more information about DHCP and DHCP options.

DNS Client

Your AlliedWare Plus™ device has a DNS Client that is enabled automatically when you add a DNS server to your device. This client allows you to use domain names instead of IP addresses when using commands on your device.

To add a DNS server to the list of servers that the device sends DNS queries to, use the command:

```
awplus(config)# ip name-server <ip-addr>
```

To check the list of servers that the device sends DNS queries to, use the command:

```
awplus# show ip name-server
```

To add a default domain name used to append to DNS requests, use the command:

```
awplus(config)# ip domain-name <domain-name>
```

For example, to use DNS to match hostnames to your internal network "example.net", use the command:

```
awplus(config)# ip domain-name example.net
```

If you then use the command **ping host2**, your device sends a DNS request for host2.example.net. To check the domain name configured with this command, use the command:

```
awplus# show ip domain-name
```

Alternatively you can create a list of domain names that your device will try in turn by using the command:

```
awplus(config)# ip domain-list <domain-name>
```

For example, to use DNS to match incomplete hostnames to the top level domains ".com", and ".net", use the commands:

```
awplus(config)# ip domain-list .com
awplus(config)# ip domain-list .net
```

If you then use the command **ping alliedtelesis**, your device sends a DNS request for alliedtelesis.com and if no match was found your device would then try alliedtelesis.net. To check the entries in the domain list, use the command:

```
awplus# show ip domain-list
```

To disable the DNS client on your device, use the command:

```
awplus(config)# no ip domain-lookup
```

To check the status of the DNS Client on your device, and the configured servers and domain names, use the command:

```
awplus# show hosts
```

DNS Relay

Enabling DNS Relay your switch provides the capability for it act as a local virtual DNS server. It can then service DNS lookup repetitive requests sent to it from local hosts. Acting as a DNS Relay the switch will usually relay the requests to an external, or upstream, DNS server. By default, DNS Relay is disabled.Optionally, DNS name resolver caching may be enabled on the DNS Relay, which can provide some lookup speed advantage and avoid unnecessary repeated requests to external DNS servers. By default, DNS caching is disabled.

When the DNS Relay name resolver cache is enabled on your switch, the switch will maintain a cache of recently used mappings between domain names and IP addresses so that other identical requests can be responded to without further reference to an external, or upstream DNS server. When the switch receives a DNS query from a client the switch will attempt to match the request with entries in this cache. If the switch does not have this address cached, it forwards the request upwards through the hierarchy of DNS servers for resolution. The DNS cache has a limited size, and times out entries after a specified period of up to 60 minutes.

The relaying of DNS queries is required for use in networks where the DNS server and the clients connected to the switch are on different subnets and do not know how to reach each other.

DNS Relay uses the DNS server list configured by the **ip name-server** command to forward DNS query packets. To enable DNS Relay you need to configure the list of servers that the device sends DNS queries to and then enable DNS forwarding, as shown in the following example for a DNS server with an IPv4 address:

```
awplus# configure terminal
awplus(config)# ip name-server 192.168.1.1
awplus(config)# ip name-server 192.168.1.2
awplus(config)# ip dns forwarding
```

Note both IPv4 and IPv6 support DNS record types. IPv4 and IPv6 are supported in DNS name-to-address and DNS address-to-name lookup processes. Specifying a name server and enabling DNS forwarding maps both IPv4 and IPv6 addresses. You can configure DNS Relay to use IPv6 addresses using the same commands used to configure DNS Relay to use IPv4 addresses, as shown in the following example:

```
awplus# configure terminal
awplus(config)# ip name-server 2001:0db8:010d::1
awplus(config)# ip name-server 2001:0db8:010d::2
awplus(config)# ip dns forwarding
```

You can then configure DNS Relay behavior with the following commands:

To set the number of times the switch will retry to forward DNS queries, use the command:

```
awplus(config)# ip dns forwarding retry <0-100>
```

To set the number of seconds to wait for a response, use the command:

```
awplus(config)# ip dns forwarding timeout <0-3600>
```

To set the DNS forwarding dead-time period in seconds, use the command:

```
awplus(config)# ip dns forwarding dead-time <60-43200>
```

At the dead-time period set, the switch stops sending requests to an unresponsive server.

To set the interface to use for forwarding and receiving DNS queries, use the command:

```
awplus(config)# ip dns forwarding source-interface
                <interface-name>
```

To specify the DNS Relay name resolver cache size and lifetime, use the command:

```
awplus(config)# ip dns forwarding cache [size <0-1000>]
                [timeout <60-3600>]
```

To remove entries from the DNS Relay name resolver cache, use the command:

```
awplus(config)# clear ip dns forwarding cache
```

Information which may be useful for troubleshooting DNS Relay is available using the DNS Relay debugging function. To enable DNS Relay debugging, use the command:

```
awplus# debug ip dns forwarding
```

To display the status of DNS Relay, use the command:

```
awplus# show ip dns forwarding
```

To display the status of DNS Relay name servers, use the command:

```
awplus# show ip dns forwarding server
```

To display the DNS Relay name resolver cache, use the command:

```
awplus# show ip dns forwarding cache
```

DHCP Options

When your device is using its DHCP client for an interface, it can receive the following DHCP options from the DHCP server:

- Option 6 - a list of DNS servers. This list appends to the DNS servers set on your device with the **ip name-server** command.
- Option 15 - a domain name used to resolve host names. This option replaces the domain name set with the **ip domain-name** command.

See **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction** for more information about DHCP and DHCP options.

Internet Control Message Protocol (ICMP)

The Internet Control Message Protocol (ICMP) allows networking devices to send information and control messages to other devices or hosts. Your device implements all non-obsolete ICMP functions.

The following table lists the ICMP messages implemented by your device.

ICMP Message Type	Device Response
Echo reply (0)	This is used to implement the ping command. Your device sends out an echo reply in response to an echo request.
Destination unreachable (3)	This message is sent when your device drops a packet because it did not have a route to the destination.
Redirect (5)	Your device issues this message to inform a local host that its target is located on the same LAN (no routing is required) or when it detects a host using a non-optimal route (usually because a link has failed or changed its status). For example, if your device receives a packet destined to its own MAC address, but with a destination IP address of another host in the local subnet, it returns an ICMP redirect to the originating host. ICMP redirects are disabled on interfaces on which local proxy ARP is enabled.
Echo request (8)	This is related to echo replies. If your device receives an echo request, it sends an echo reply. If you enter the ping command, your device generates echo requests.
Router Advertisements (10)	These are Router Discovery Protocol messages. If Router Discovery is enabled, your device sends these to announce the IP addresses of the sending interface.
Time to Live Exceeded (11)	If the TTL field in a packet falls to zero, your device sends this message.This occurs when there are too many hops in the path that a packet is traversing.

ICMP messages are enabled on all interfaces by default. You can control the flow of ICMP messages across different interfaces using the **access-list** commands. See **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands** and **Chapter 59, IPv4 Software Access Control List (ACL) Commands**.

Checking IP Connections

To verify connections between networks and network devices, use the ping (Packet Internet Groper) and trace route functions on your device.

Ping

Ping tests the connectivity between two network devices to determine whether each network device can “see” the other device. Echo request packets are sent to the destination addresses and responses are displayed on the console.

If you can ping the end destination, then the physical, Layer 2 and Layer 3 links are functioning, and any difficulties are in the network or higher layers.

If pinging the end destination fails, use traceroute to discover the point of failure in the route to the destination.

To ping a device, use the command:

```
awplus# ping {<hostname>|<ipaddr>}
```

Traceroute

You can use traceroute to discover the route that packets pass between two systems running the IP protocol. Traceroute sends an initial UDP packets with the Time To Live (TTL) field in the IP header set starting at 1. The TTL field is increased by one for every subsequent packet sent until the destination is reached. Each hop along the path between two systems responds with a TTL exceeded packet (ICMP type 11) and from this the path is determined.

To use traceroute, use the command:

```
awplus# traceroute {<ip-addr>|<hostname>}
```

Enter either the hostname or the IP address of the device you are trying to reach.

IP Helper

The IP Helper feature allows the switch to receive UDP broadcasts on one subnet, and forward them as broadcasts or unicasts into another subnet, so a client can use an application which uses UDP broadcast (such as Net-BIOS) when the client and server are located in different subnets. The IP Helper feature forwards UDP broadcast network traffic to specific hosts on another subnet and/or to the broadcast address of another subnet.

When the IP Helper feature is enabled on a VLAN interface, the UDP broadcast packets received on the interface are processed for forwarding out through another interface into another subnet. Depending on the nature of the ip-helper addresses configured, the UDP broadcasts will be unicast forwarded to a single host in the destination subnet, or unicast forwarded to multiple hosts in the destination subnet, or broadcast to the broadcast address of the destination subnet. Not all UDP broadcasts will be forwarded when IP Helper is configured. The set of broadcasts to be forwarded can be defined by specifying the destination UDP port(s) of the packets you wish to forward.

The command to enable the forwarding of UDP broadcasts received on a given interface is **ip helper-address** (entered in interface configuration mode). The **ip forward-protocol udp** command specifies types of broadcast packets to forward.

Multiple different destination addresses can be specified by using multiple instances of the **ip helper-address** command under the same interface. If a destination address is specified that is actually the broadcast address of one of the subnets directly connected to the switch, then the UDP packets will be forwarded as broadcasts onto that subnet.

Likewise, multiple different types of UDP packet can be specified for forwarding by specifying multiple different destination ports using the **ip forward-protocol udp** command.

Note

The types of UDP broadcast packets that the switch will forward are **only** those specified by the **ip forward-protocol** command(s). The IP Helper process does not forward any other UDP packet types by default.

IP Directed Broadcast

IP directed-broadcast is enabled and disabled per VLAN interface. When enabled a directed broadcast packet is forwarded to an enabled VLAN interface if received on another subnet.

An IP directed broadcast is an IP packet whose destination address is a broadcast address for some IP subnet, but originates from a node that is not itself part of that destination subnet. When a directed broadcast packet reaches a switch that is directly connected to its destination subnet, the packet is flooded as a broadcast on the destination subnet.

The **ip directed-broadcast** command controls the flooding of directed broadcasts when they reach target subnets. The command affects the final transmission of the directed broadcast on its destination subnet. It does not affect the transit unicast routing of IP directed broadcasts. If directed broadcast is enabled for an interface, incoming directed broadcast IP packets intended for the subnet assigned to interface will be flooded as broadcasts on that subnet.

If the **no ip directed-broadcast** command is configured for an interface, directed broadcasts destined for the subnet where the interface is attached will be dropped instead of broadcast.

Chapter 29: IP Addressing and Protocol Commands

Introduction 29.2
Command List 29.3
arp-mac-disparity 29.3
arp-aging-timeout 29.4
arp (IP address MAC) 29.5
arp log 29.6
arp opportunistic-nd 29.9
clear arp-cache 29.10
clear ip dns forwarding cache 29.10
debug ip dns forwarding 29.11
debug ip packet interface 29.11
ip address 29.13
ip directed-broadcast 29.15
ip dns forwarding 29.16
ip dns forwarding cache 29.17
ip dns forwarding dead-time 29.18
ip dns forwarding retry 29.19
ip dns forwarding source-interface 29.20
ip dns forwarding timeout 29.21
ip domain-list 29.22
ip domain-lookup 29.23
ip domain-name 29.24
ip forward-protocol udp 29.25
ip gratuitous-arp-link 29.27
ip helper-address 29.28
ip local-proxy-arp 29.30
ip name-server 29.31
ip proxy-arp 29.32
ip redirects 29.33
optimistic-nd 29.34
ping 29.35
show arp 29.36
show debugging ip dns forwarding 29.37
show debugging ip packet 29.38
show hosts 29.39
show ip dns forwarding 29.40
show ip dns forwarding cache 29.41
show ip dns forwarding server 29.42
show ip domain-list 29.43
show ip domain-name 29.43
show ip interface 29.44
show ip name-server 29.45
show ip sockets 29.46
show ip traffic 29.48
tcpdump 29.54
traceroute 29.55

Introduction

This chapter provides an alphabetical reference of commands used to configure the following protocols:

- Address Resolution Protocol (ARP)
- Domain Name Service (DNS)

For more information see **Chapter 28, Internet Protocol (IP) Addressing and Protocols**.

Command List

arp-mac-disparity

Use this command in Interface Configuration mode for a VLAN interface to enable the reception of ARP packets that contain a multicast MAC address in the sender field.

By default, ARP packets that contain a multicast MAC address in the sender field are dropped. The **no** variant of this command reverts to the default behavior.

Syntax

```
arp-mac-disparity
no arp-mac-disparity
```

Default ARP disparity is disabled. ARP packets with a multicast MAC address in the sender field are dropped.

Mode Interface Configuration for a VLAN interface.

Usage Normally, it is invalid for an ARP request to resolve a multicast MAC address. By default, ARP replies with a multicast MAC addresses are not learned. This command allows control over the learning of dynamic ARPs that resolve to a multicast MAC address.

ARP-MAC disparity may need to be enabled to support multicast network load balancing. The **arp-mac-disparity** command allows ARP replies quoting multicast MAC addresses to be accepted and learned. No **no arp-mac-disparity** command reverts to default behavior.

If the ARP-MAC disparity feature is enabled, then the switch sends traffic to a single port as specified by the ARP entry.

Examples To enable ARP MAC disparity on interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# arp-mac-disparity
```

To disable ARP MAC disparity on interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no arp-mac-disparity
```

Related Commands

clear arp-cache
show arp

arp-aging-timeout

This command sets a timeout period on dynamic ARP entries associated with a specific interface. If your device stops receiving traffic for the host specified in a dynamic ARP entry, it deletes the ARP entry from the ARP cache after this timeout is reached.

Your device times out dynamic ARP entries to ensure that the cache does not fill with entries for hosts that are no longer active. Static ARP entries are not aged or automatically deleted.

By default the time limit for dynamic ARP entries is 300 seconds on all interfaces.

The **no** variant of this command sets the time limit to the default of 300 seconds.

Syntax

```
arp-aging-timeout <0-432000>
no arp-aging timeout
```

Parameter	Description
<0-432000>	The timeout period in seconds.

Default 300 seconds (5 minutes)

Mode Interface Configuration for a VLAN interface.

Example To set the ARP entries on interface `vlan30` to time out after two minutes, use the commands:

```
awplus(config)# interface vlan30
awplus(config-if)# arp-aging-timeout 120
```

Related Commands

clear arp-cache
show arp

arp (IP address MAC)

This command adds a static ARP entry to the ARP cache. This is typically used to add entries for hosts that do not support ARP or to speed up the address resolution function for a host. The ARP entry must not already exist. Use the **alias** parameter to allow your device to respond to ARP requests for this IP address.

The **no** variant of this command removes the static ARP entry. Use the **clear arp-cache** command on page 29.10 to remove the dynamic ARP entries in the ARP cache.

Syntax

```
arp <ip-addr> <mac-address> [<port-number>] [alias]
no arp <ip-addr>
```

Parameter	Description
<ip-addr>	IPv4 address of the device you are adding as a static ARP entry.
<mac-address>	MAC address of the device you are adding as a static ARP entry, in hexadecimal notation with the format HHHH.HHHH.HHHH.
<port-number>	The port number associated with the IP address. Specify this when the IP address is part of a VLAN.
alias	Allows your device to respond to ARP requests for the IP address. Proxy ARP must be enabled on the interface before using this parameter.

Mode Global Configuration

Examples To add the IP address 10.10.10.9 with the MAC address 0010.2533.4655 into the ARP cache, and have your device respond to ARP requests for this address, use the commands:

```
awplus# configure terminal
awplus(config)# arp 10.10.10.9 0010.2355.4566 alias
```

Related Commands

clear arp-cache
ip proxy-arp
show arp

arp log

This command enables the logging of dynamic and static ARP entries in the ARP cache. The ARP cache contains mappings of switch ports, VLAN IDs, and IP addresses to physical MAC addresses for hosts.

This command can display the MAC addresses in the ARP log either using the default hexadecimal notation (HHHH.HHHH.HHHH), or using the IEEE standard hexadecimal notation (HH-HH-HH-HH-HH-HH).

Use the **no** variant of this command to disable the logging of dynamic and static ARP entries in the ARP cache.

Syntax

```
arp log [mac-address-format ieee]

no arp log [mac-address-format ieee]
```

Parameter	Description
`mac-address-format ieee`	Display the MAC address in hexadecimal notation with the standard IEEE format (HH-HH-HH-HH-HH-HH), instead of displaying the MAC address with the default hexadecimal format (HHHH.HHHH.HHHH).

Default The ARP logging feature is disabled by default.

Mode Global Configuration

Usage You have the option to change how the MAC address is displayed in the ARP log message, to use the default hexadecimal notation (HHHH.HHHH.HHHH), or the IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH) when you apply the **mac-address-format ieee** parameter.

Enter the **arp log** command without the optional **mac-address-format ieee** parameter specified for MAC addresses in the ARP log output to use the default hexadecimal notation (HHHH.HHHH.HHHH).

Enter the **arp log mac-address-format ieee** command for MAC addresses in the ARP log output to use the IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH).

Use the **no** variant of this command (**no arp log**) without the optional **mac-address-format ieee** parameter specified to disable ARP logging on the switch

Use the **no** variant of this command with the optional **mac-address-format ieee** parameter specified (**no arp log mac-address-format ieee**) to disable IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH) and revert to the default hexadecimal notation (HHHH.HHHH.HHHH) for MAC addresses in the ARP log output.

To display ARP log messages use the **show log | include ARP_LOG** command.

Examples To enable ARP logging and use the default hexadecimal notation (HHHH.HHHH.HHHH), use the following commands:

```
awplus# configure terminal
awplus(config)# arp log
```

To disable ARP logging on the switch of MAC addresses displayed using the default hexadecimal notation (HHHH.HHHH.HHHH), use the following commands:

```
awplus# configure terminal
awplus(config)# no arp log
```

To enable ARP logging and to specify that the MAC address in the log message is displayed in the standard IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH), use the following commands:

```
awplus# configure terminal
awplus(config)# arp log mac-address-format ieee
```

To disable ARP logging on the switch of MAC addresses displayed using the standard IEEE format hexadecimal notation (HH-HH-HH-HH-HH-HH), and revert to the use of the default hexadecimal notation (HHHH.HHHH.HHHH) instead, use the following commands:

```
awplus# configure terminal
awplus(config)# no arp log mac-address-format ieee
```

To display ARP log messages, use following command:

```
awplus# show log | include ARP_LOG
```

Output Below is example output from the **show log | include ARP_LOG** command after enabling ARP logging displaying default hexadecimal notation MAC addresses (HHHH.HHHH.HHHH) using the **arp log** command.

```
awplus#configure terminal
awplus(config)#arp log
awplus(config)#exit
awplus#show log | include ARP_LOG
2010 Apr  6 06:21:01 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 add
0013.4078.3b98 (192.168.2.4)
2010 Apr  6 06:22:30 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 del
0013.4078.3b98 (192.168.2.4)
2010 Apr  6 06:23:26 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 add
0030.940e.136b (192.168.2.20)
2010 Apr  6 06:23:30 user.notice awplus IMISH[1830]: show log | include ARP_LOG
```

Below is example output from the **show log | include ARP_LOG** command after enabling ARP logging displaying IEEE standard format hexadecimal notation MAC addresses (HH-HH-HH-HH-HH-HH) using the **arp log mac-address format ieee** command.

Figure 29-1: Example output from the show log | include ARP_LOG command

```
awplus#configure terminal
awplus(config)#arp log mac-address-format ieee
awplus(config)#exit
awplus#show log | include ARP_LOG
2010 Apr  6 06:25:28 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 add 00-
17-9a-b6-03-69 (192.168.2.12)
2010 Apr  6 06:25:30 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 add 00-
03-37-6b-a6-a5 (192.168.2.10)
2010 Apr  6 06:26:53 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 del 00-
30-94-0e-13-6b (192.168.2.20)
2010 Apr  6 06:27:31 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 del 00-
17-9a-b6-03-69 (192.168.2.12)
2010 Apr  6 06:28:09 user.notice awplus HSL[1007]: ARP_LOG port1.0.6 vlan1 del 00-
03-37-6b-a6-a5 (192.168.2.10)
2010 Apr  6 06:28:14 user.notice awplus IMISH[1830]: show log | include ARP_LOG
```

Below are the parameters in output of the **show log | include ARP_LOG** command with an ARP log message format of *<ARP_LOG> <port number> <VLAN ID> <Operation> <MAC> <IP>* after *<date> <time> <severity> <hostname> <program name>* information.

Table 29-1: Parameters in output of the show log | include ARP_LOG command

Parameter	Description
<ARP_LOG>	Indicates ARP log entry information follows *<date> <time> <severity> <hostname> <program name>* log information.
<port number>	Indicates switch port number for the ARP log entry.
<VLAN ID>	Indicates the VLAN ID for the ARP log entry.
<Operation>	Indicates 'add' if the ARP log entry displays an ARP addition. Indicates 'del' if the ARP log entry displays an ARP deletion.
<MAC>	Indicates the MAC address for the ARP log entry, either in the default hexadecimal notation (HHHH.HHHH.HHHH) or in the IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH) as specified with the **arp log** or the **arp log mac-address-format ieee** command.
<IP>	Indicates the IP address for the ARP log entry.

Validation Commands **show running-config**

Related Commands **show log**

arp opportunistic-nd

This command changes the behavior for unsolicited ARP packet forwarding on the switch.

Use this command to enable opportunistic neighbor discovery for the global ARP cache.

Use the **no** variant of this command to disable opportunistic neighbor discovery for the global ARP cache.

Syntax

```
arp opportunistic-nd
no arp opportunistic-nd
```

Default Opportunistic neighbor discovery is disabled by default.

Mode Global Configuration

Usage When opportunistic neighbor discovery is enabled, the switch will reply to any received unsolicited ARP packets (but not gratuitous ARP packets). The source MAC address for the unsolicited ARP packet is added to the ARP cache, so the switch forwards the ARP packet. When opportunistic neighbor discovery is disabled, the source MAC address for the ARP packet is not added to the ARP cache, so the ARP packet is not forwarded by the switch.

Note this command enables or disables opportunistic neighbor discovery for a VRF lite instance if the **VRF lite** parameter and an instance name are applied. If a VRF lite instance is not specified, then opportunistic neighbor discovery is enabled or disabled for switch ports configured for IPv4.

Examples To enable opportunistic neighbor discovery for the global ARP cache, enter:

```
awplus# configure terminal
awplus(config)# arp opportunistic-nd
```

To disable opportunistic neighbor discovery for the global ARP cache, enter:

```
awplus# configure terminal
awplus(config)# no arp opportunistic-nd
```

Related Commands **ipv6 opportunistic-nd**
show arp

Validation Commands **show running-config interface**

clear arp-cache

This command deletes dynamic ARP entries from the ARP cache. You can optionally specify the IPv4 address of an ARP entry to be cleared from the ARP cache.

Syntax `clear arp-cache [<ip-address>]`

Mode Privileged Exec

Usage To display the entries in the ARP cache, use the **show arp** command. To remove static ARP entries, use the no variant of the **arp (IP address MAC)** command on page 29.5.

Example To clear all dynamic ARP entries, use the command:

```
awplus# clear arp-cache
```

To clear all dynamic ARP entries associated with the IPv4 address 192.168.1.1, use the command:

```
awplus# clear arp-cache 192.168.1.1
```

Related Commands **arp-mac-disparity**
arp (IP address MAC)
show arp

clear ip dns forwarding cache

Use this command to clear the DNS Relay name resolver cache.

Syntax `clear ip dns forwarding cache`

Mode Privileged Exec

Examples To clear all cached data, use the command:

```
awplus# clear ip dns forwarding cache
```

Related Commands **ip dns forwarding cache**

debug ip dns forwarding

Use this command to enable DNS Relay debugging.

Use the **no** variant of this command to disable DNS Relay debugging.

Syntax

```
debug ip dns forwarding
no debug ip dns forwarding
```

Default DNS Relay debugging is disabled by default.

Mode Privileged Exec

Examples To enable DNS forwarding debugging, use the commands:

```
awplus# debug ip dns forwarding
```

To disable DNS forwarding debugging, use the commands:

```
awplus# no debug ip dns forwarding
```

Related Commands

ip dns forwarding
show debugging ip dns forwarding

debug ip packet interface

The **debug ip packet interface** command enables IP packet debug and is controlled by the **terminal monitor** command.

If the optional **icmp** keyword is specified then ICMP packets are shown in the output.

The **no** variant of this command disables the **debug ip interface** command.

Syntax

```
debug ip packet interface {<interface-name>|all}
   [address <ip-address>|verbose|hex|arp|udp|tcp|icmp]
no debug ip packet interface [<interface-name>]
```

Parameter	Description
<interface>	Specify a single Layer 3 interface name (not a range of interfaces) This keyword can be specified as either all or as a single Layer 3 interface to show debugging for either all interfaces or a single interface.
`all`	Specify all Layer 3 interfaces on the switch.
<ip-address>	Specify an IPv4 address. If this keyword is specified, then only packets with the specified IP address as specified in the ip-address placeholder are shown in the output.

Parameter	Description
`verbose`	Specify **verbose** to output more of the IP packet. If this keyword is specified then more of the packet is shown in the output.
`hex`	Specify **hex** to output the IP packet in hexadecimal. If this keyword is specified, then the output for the packet is shown in hex.
`arp`	Specify **arp** to output ARP protocol packets. If this keyword is specified, then ARP packets are shown in the output.
`udp`	Specify **udp** to output UDP protocol packets. If this keyword is specified then UDP packets are shown in the output.
`tcp`	Specify **tcp** to output TCP protocol packets. If this keyword is specified, then TCP packets are shown in the output.
`icmp`	Specify **icmp** to output ICMP protocol packets. If this keyword is specified, then ICMP packets are shown in the output.

Mode Privileged Exec and Global Configuration

Examples To turn on ARP packet debugging on `vlan1`, use the command:

```
awplus# debug ip packet interface vlan1 arp
```

To turn on all packet debugging on all interfaces on the switch, use the command:

```
awplus# debug ip packet interface all
```

To turn on TCP packet debugging on `vlan1` and IP address `192.168.2.4`, use the command:

```
awplus# debug ip packet interface vlan1 address 192.168.2.4
       tcp
```

To turn off IP packet interface debugging on all interfaces, use the command:

```
awplus# no debug ip packet interface
```

To turn off IP packet interface debugging on interface `vlan2`, use the command:

```
awplus# no debug ip packet interface vlan2
```

Related Commands
no debug all
show debugging ip dns forwarding
tcpdump
terminal monitor

ip address

This command sets a static IP address on an interface. To set the primary IP address on the interface, specify only **ip address <*ip-address/m*>**. This overwrites any configured primary IP address. To add additional IP addresses on this interface, use the **secondary** parameter. You must configure a primary address on the interface before configuring a secondary address.

Note Use **show running-config** interface not **show ip interface brief** when you need to view a secondary address configured on an interface. **show ip interface brief** will only show the primary address not a secondary address for an interface.

The **no** variant of this command removes the IP address from the interface. You cannot remove the primary address when a secondary address is present.

Syntax

```
ip address <ip-addr/prefix-length> [secondary] [label <label>]
no ip address <ip-addr/prefix-length> [secondary]
no ip address
```

Parameter	Description
<ip-addr/prefix-length>	The IPv4 address and prefix length you are assigning to the interface.
`secondary`	Secondary IP address.
`label`	Adds a user-defined description of the secondary IP address.
<label>	A user-defined description of the secondary IP address. Valid characters are any printable character and spaces.

Mode Interface Configuration for a VLAN interface or a local loopback interface.

Examples To add the primary IP address `10.10.10.50/24` to the interface `vlan3`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip address 10.10.10.50/24
```

To add the secondary IP address `10.10.11.50/24` to the same interface, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip address 10.10.11.50/24 secondary
```

To add the IP address `10.10.11.50/24` to the local loopback interface `lo`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface lo
awplus(config-if)# ip address 10.10.11.50/24
```

Related Commands

interface (to configure)
show ip interface
show running-config interface

ip directed-broadcast

Use this command to enable flooding of directed broadcast packets into a directly connected subnet. If this command is configured on a VLAN interface, then directed broadcasts received on other VLAN interfaces, destined for the subnet on this VLAN, will be flooded to the subnet broadcast address of this VLAN.

Use the **no** variant of this command to disable **ip directed-broadcast**. When this feature is disabled using the **no** variant of this command, directed broadcasts are not forwarded.

Syntax

```
ip directed-broadcast
no ip directed-broadcast
```

Default The **ip directed-broadcast** command is disabled by default.

Mode Interface Configuration for a VLAN interface or a local loopback interface.

Usage IP directed-broadcast is enabled and disabled per VLAN interface. When enabled a directed broadcast packet is forwarded to an enabled VLAN interface if received on another subnet.

An IP directed broadcast is an IP packet whose destination address is a broadcast address for some IP subnet, but originates from a node that is not itself part of that destination subnet. When a directed broadcast packet reaches a switch that is directly connected to its destination subnet, that packet is flooded as a broadcast on the destination subnet.

The **ip directed-broadcast c**ommand controls the flooding of directed broadcasts when they reach target subnets. The command affects the final transmission of the directed broadcast on its destination subnet. It does not affect the transit unicast routing of IP directed broadcasts. If directed broadcast is enabled for an interface, incoming directed broadcast IP packets intended for the subnet assigned to interface will be flooded as broadcasts on that subnet.

If the **no ip directed-broadcast** command is configured for an interface, directed broadcasts destined for the subnet where the interface is attached will be dropped instead of broadcast.

Examples To enable **ip directed-broadcast**, to flood broadcast packets out via the `vlan2` interface, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip directed-broadcast
```

To disable **ip directed-broadcast,** disabling the flooding of broadcast packets via `vlan2`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip directed-broadcast
```

Related Commands

ip forward-protocol udp
ip helper-address
show running-config

ip dns forwarding

Use this command to enable DNS Relay, the forwarding of incoming DNS queries for IP hostname-to-address translation.

Use the **no** variant of this command to disable the forwarding of incoming DNS queries for IP hostname-to-address translation.

Syntax

```
ip dns forwarding
no ip dns forwarding
```

Default The forwarding of incoming DNS query packets is disabled by default.

Mode Global Configuration

Usage See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To enable the forwarding of incoming DNS query packets, use the commands:

```
awplus# configure terminal
awplus(config)# ip dns forwarding
```

To disable the forwarding of incoming DNS query packets, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dns forwarding
```

Related Commands

clear ip dns forwarding cache
debug ip dns forwarding
ip dns forwarding cache
ip dns forwarding dead-time
ip dns forwarding retry
ip dns forwarding source-interface
ip dns forwarding timeout
ip name-server
show ip dns forwarding
show ip dns forwarding cache
show ip dns forwarding server

ip dns forwarding cache

Use this command to set the DNS Relay name resolver cache size and cache entry lifetime period. The DNS Relay name resolver cache stores the mappings between domain names and IP addresses.

Use the **no** variant of this command to set the default DNS Relay name resolver cache size and cache entry lifetime period.

Note that the lifetime period of the cache entry can be overwritten by the time-out period of the DNS reply from the DNS server if the time-out period of the DNS reply from the DNS server is smaller than the configured time-out period. The time-out period of the cache entry will only be used when the time-out period of the DNS reply from the DNS server is bigger than the time-out period configured on the device.

Syntax

```
ip dns forwarding cache [size <0-1000>] [timeout <60-3600>]
no ip dns forwarding cache [size|timeout]
```

Parameter	Description
<0-1000>	Number of entries in the DNS Relay name resolver cache.
<60-3600>	Timeout value in seconds.

Default The default cache size is 0 (no entries) and the default lifetime is 1800 seconds.

Mode Global Configuration

Usage See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set the cache size to `10` entries and the lifetime to `500` seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip dns forwarding cache size 10 time 500
```

To set the cache size to the default, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dns forwarding cache size
```

Related Commands

clear ip dns forwarding cache
debug ip dns forwarding
ip dns forwarding
show ip dns forwarding
show ip dns forwarding cache

ip dns forwarding dead-time

Use this command to set the time period in seconds when the switch stops sending any DNS requests to an unresponsive server and all retries set using **ip dns forwarding retry** are used. This time period is the DNS forwarding dead-time. The switch stops sending DNS requests at the DNS forwarding dead-time configured and when all of the retries are used.

Use the **no** variant of this command to restore the default DNS forwarding dead-time value of 3600 seconds.

Syntax

```
ip dns forwarding dead-time <60-43200>
no ip dns forwarding retry
```

Parameter	Description
<60-43200>	Set the DNS forwarding dead-time in seconds. At the dead-time set, the switch stops sending DNS requests to an unresponsive server.

Default The default time to stop sending DNS requests to an unresponsive server is 3600 seconds.

Mode Global Configuration

Usage See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings. See the **ip dns forwarding retry** command used with this command.

Examples To set the DNS forwarding retry count to `50` and to set the DNS forwarding dead-time to `1800` seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip dns forwarding dead-time 1800
awplus(config)# ip dns forwarding retry 50
```

To reset the DNS retry count to the default of `2` and the DNS forwarding dead-time to the default of 3600, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dns forwarding dead-time
awplus(config)# no ip dns forwarding retry
```

Related Commands

debug ip dns forwarding
ip dns forwarding
ip dns forwarding retry
show ip dns forwarding
show ip dns forwarding server

ip dns forwarding retry

Use this command to set the number of times DNS Relay will retry to forward DNS queries. The switch stops sending DNS requests to an unresponsive server at the time set using the **ip dns forwarding dead-time** command and when all of the retries are used.

Use the **no** variant of this command to set the number of retries to the default of 2.

Syntax

```
ip dns forwarding retry <0-100>
no ip dns forwarding retry
```

Parameter	Description
<0-100>	Set the number of times DNS Relay will retry to forward a DNS query.

Default The default number of retries is 2 DNS requests to an unresponsive server.

Mode Global Configuration

Usage See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings. See the **ip dns forwarding dead-time** command used with this command.

Examples To set the DNS forwarding retry count to `50` and to set the DNS forwarding dead-time to `1800` seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip dns forwarding retry 50
awplus(config)# ip dns forwarding dead-time 1800
```

To reset the DNS retry count to the default of `2` and the DNS forwarding dead-time to the default of `3600` seconds, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dns forwarding retry
awplus(config)# no ip dns forwarding dead-time
```

Related Commands

debug ip dns forwarding
ip dns forwarding
ip dns forwarding dead-time
show ip dns forwarding

ip dns forwarding source-interface

Use this command to set the interface to use for forwarding and receiving DNS queries.

Use the **no** variant of this command to unset the interface used for forwarding and receiving DNS queries.

Syntax

```
ip dns forwarding source-interface <interface-name>
no ip dns forwarding source-interface
```

Parameter	Description
<interface-name>	An alphanumeric string that is the interface name.

Default The default is that no interface is set and the switch selects the appropriate source IP address automatically.

Mode Global Configuration

Usage See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set `vlan1` as the source interface for relayed DNS queries, use the commands:

```
awplus# configure terminal
awplus(config)# ip dns forwarding source-interface vlan1
```

To clear the source interface for relayed DNS queries, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dns forwarding source-interface
```

Related Commands

debug ip dns forwarding
ip dns forwarding
show ip dns forwarding

ip dns forwarding timeout

Use this command to set the time period for the DNS Relay to wait for a DNS response.

Use the **no** variant of this command to set the time period to wait for a DNS response to the default of 3 seconds.

Syntax

```
ip dns forwarding timeout <0-3600>
no ip dns forwarding timeout
```

Parameter	Description
<0-3600>	Timeout value in seconds.

Default The default timeout value is 3 seconds.

Mode Global Configuration

Usage See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a a database of hostname-to-address mappings.

Examples To set the timeout value to `12` seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip dns forwarding timeout 12
```

To set the timeout value to the default of `3` seconds, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dns forwarding timeout
```

Related Commands

debug ip dns forwarding
ip dns forwarding
show ip dns forwarding

ip domain-list

This command adds a domain to the DNS list. Domain are appended to incomplete host names in DNS requests. Each domain in this list is tried in turn in DNS lookups. This list is ordered so that the first entry you create is checked first.

The **no** variant of this command deletes a domain from the list.

Syntax

```
ip domain-list <domain-name>
no ip domain-list <domain-name>
```

Parameter	Description
<domain-name>	Domain string, for example “company.com”.

Mode Global Configuration

Usage If there are no domains in the DNS list, then your device uses the domain specified with the **ip domain-name** command. If any domain exists in the DNS list, then the device does not use the domain set using the **ip domain-name** command.

See **“Domain Name System (DNS)” on page 28.8** for introductory information about DNS.

See **“DNS Client” on page 28.9** for information about DNS Client configuration commands.

Example To add the domain `example.net` to the DNS list, use the following commands:

```
awplus# configure terminal
awplus(config)# ip domain-list example.net
```

Related Commands

ip domain-lookup
ip domain-name
show ip domain-list

ip domain-lookup

This command enables the DNS client on your device. This allows you to use domain names instead of IP addresses in commands. The DNS client resolves the domain name into an IP address by sending a DNS inquiry to a DNS server, specified with the **ip name-server** command.

The **no** variant of this command disables the DNS client. The client will not attempt to resolve domain names. You must use IP addresses to specify hosts in commands.

Syntax

```
ip domain-lookup
no ip domain-lookup
```

Mode Global Configuration

Usage The client is enabled by default. However, it does not attempt DNS inquiries unless there is a DNS server configured.

See **"DNS Client" on page 28.9** for information about DNS Client configuration commands.

Examples To enable the DNS client on your device, use the following commands:

```
awplus# configure terminal
awplus(config)# ip domain-lookup
```

To disable the DNS client on your device, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip domain-lookup
```

Related Commands

ip domain-list
ip domain-name
ip name-server
show hosts
show ip name-server

ip domain-name

This command sets a default domain for the DNS. The DNS client appends this domain to incomplete host-names in DNS requests.

The **no** variant of this command removes the domain-name previously set by this command.

Syntax

```
ip domain-name <domain-name>
no ip domain-name <domain-name>
```

Parameter	Description
<domain-name>	Domain string, for example “company.com”.

Mode Global Configuration

Usage If there are no domains in the DNS list (created using the **ip domain-list** command) then your device uses the domain specified with this command. If any domain exists in the DNS list, then the device does not use the domain configured with this command.

See **“DNS Client” on page 28.9** for information about DNS Client configuration commands.

When your device is using its DHCP client for an interface, it can receive Option 15 from the DHCP server. This option replaces the domain name set with this command. See **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction** for more information about DHCP and DHCP options.

Example To configure the domain name, enter the following commands:

```
awplus# configure terminal
awplus(config)# ip domain-name company.com
```

Related Commands

ip domain-list
show ip domain-list
show ip domain-name

ip forward-protocol udp

This command enables you to control which UDP broadcasts will be forwarded to the helper address(es). A UDP broadcast will only be forwarded if the destination UDP port number in the packet matches one of the port numbers specified using this command.

Refer to the IANA site (**www.iana.org**) for a list of assigned UDP port numbers for protocols to forward using **ip forward-protocol udp**.

Use the **no** variant of this command to remove a port number from the list of destination port numbers that are used as the criterion for deciding if a given UDP broadcast should be forwarded to the IP helper address(es).

Syntax

```
ip forward-protocol udp <port>
no ip forward-protocol udp <port>
```

Parameter	Description
<port>	UDP Port Number.

Default The **ip forward-protocol udp** command is not enabled by default.

Mode Global Configuration

Usage Combined with the **ip helper-address** command on page 29.28 in interface mode, the **ip forward-protocol udp** command in Global Configuration mode allows control of which protocols (destination port numbers) are forwarded. The **ip forward-protocol udp** command configures protocols for forwarding, and the **ip helper-address** command configures the destination address(es).

Note The types of UDP broadcast packets that the switch will forward are ONLY those specified by the **ip forward-protocol** command(s). There are no other UDP packet types that the IP helper process forwards by default.

Note The **ip forward-protocol udp** command does not support BOOTP / DHCP Relay. The **ip dhcp-relay** command must be used instead. For this reason, you may not configure UDP ports 67 and 68 with the **ip forward-protocol udp** command.
See **"DHCP Relay Agent Introduction" on page 89.9** for information about DHCP Relay.

Examples To configure forwarding of packets on a UDP port, use the following commands:

```
awplus# configure terminal
awplus(config)# ip forward-protocol udp <port>
```

To delete a UDP port from the UDP ports that the switch forwards, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip forward-protocol udp <port>
```

Validation Commands **show running-config**

Related Commands **ip helper-address**
ip directed-broadcast

ip gratuitous-arp-link

This command sets the Gratuitous ARP time limit for all switchports. The time limit restricts the sending of Gratuitous ARP packets to one Gratuitous ARP packet within the time in seconds.

Note

This command specifies time between sequences of Gratuitous ARP packets, and time between individual Gratuitous ARP packets occurring in a sequence, to allow legacy support for older devices and interoperation between other devices that are not ready to receive and forward data until several seconds after linkup.

Additionally, jitter has been applied to the delay following linkup, so Gratuitous ARP packets applicable to a given port are spread over a period of 1 second so are not all sent at once. Remaining Gratuitous ARP packets in the sequence occur after a fixed delay from the first one.

Syntax

```
ip gratuitous-arp-link <0-300>

no ip gratuitous-arp-link
```

Parameter	Description
<0-300>	Specify the minimum time between sequences of Gratuitous ARPs and the fixed time between Gratuitous ARPs occurring in a sequence, in seconds. 0 disables the sending of Gratuitous ARP packets. The default is 8 seconds.

Default The default Gratuitous ARP time limit for all switchports is 8 seconds.

Mode Global Configuration

Usage Every switchport will send a sequence of 3 Gratuitous ARP packets to each VLAN that the switchport is a member of, whenever the switchport moves to the forwarding state. The first Gratuitous ARP packet is sent 1 second after the switchport becomes a forwarding switchport. The second and third Gratuitous ARP packets are each sent after the time period specified by the Gratuitous ARP time limit.

Additionally, the Gratuitous ARP time limit specifies the minimum time between the end of one Gratuitous ARP sequence and the start of another Gratuitous ARP sequence. When a link is flapping, the switchport's state is set to forwarding several times. The Gratuitous ARP time limit is imposed to prevent Gratuitous ARP packets from being sent undesirably often.

Examples To disable the sending of Gratuitous ARP packets, use the commands:

```
awplus# configure terminal
awplus(config)# ip gratuitous-arp-link 0
```

To restrict the sending of Gratuitous ARP packets to one every 20 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip gratuitous-arp-link 20
```

Validation Commands **show running-config**

ip helper-address

This command adds a forwarding destination address for IP Helper to enable forwarding of User Datagram Protocol (UDP) broadcasts on an interface.

Use the **no** variant of this command to disable the forwarding of broadcast packets to specific addresses.

Syntax

```
ip helper-address <ip-addr>
no ip helper-address <ip-addr>
```

Parameter	Description
<ip-addr>	Forwarding destination IP address for IP Helper.

Default The destination address for the **ip helper-address** command is not configured by default.

Mode Interface Configuration for a VLAN interface or a local loopback interface.

Usage Combined with the **ip forward-protocol udp** command in global configuration mode, the **ip helper-address** command in interface mode allows control of which protocols (destination port numbers) are forwarded. The **ip forward-protocol udp** command configures protocols for forwarding, and the **ip helper-address** command configures the destination address(es).

The destination address can be a unicast address or a subnet broadcast address. The UDP destination port is configured separately with the **ip forward-protocol udp** command. If multiple destination addresses are registered then UDP packets are forwarded to each IP address added to an IP Helper. Up to 32 destination addresses may be added using IP Helper.

Note The types of UDP broadcast packets that the switch will forward are ONLY those specified by the **ip forward-protocol** command(s). There are no other UDP packet types that the IP helper process forwards by default.

Note The **ip helper-address** command does not support BOOTP / DHCP Relay. The **ip dhcp-relay** command must be used instead. For this reason, you may not configure UDP ports 67 and 68 with the **ip forward-protocol** command. For information about DHCP Relay, see **"DHCP Relay Agent Introduction" on page 89.9**.

Examples The following example defines IPv4 address `192.168.1.100` as an IP Helper destination address to which to forward UDP broadcasts received on `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip helper-address 192.168.1.100
```

The following example removes IPv4 address `192.168.1.100` as an IP Helper destination address to which to forward UDP broadcasts received on `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip helper-address 192.168.1.100
```

Validation Commands **show running-config**

Related Commands **ip forward-protocol udp**
ip directed-broadcast

ip local-proxy-arp

This command allows you to stop MAC address resolution between hosts within a private VLAN edge interface. Local Proxy ARP works by intercepting ARP requests between hosts within a subnet and responding with your device's own MAC address details instead of the destination host's details. This stops hosts from learning the MAC address of other hosts within its subnet through ARP requests.

Local Proxy ARP ensures that devices within a subnet cannot send traffic that bypasses Layer 3 routing on your device. This lets you monitor and filter traffic between hosts in the same subnet, and enables you to have control over which hosts may communicate with one another.

When Local Proxy ARP is operating on an interface, your device does not generate or forward any ICMP-Redirect messages on that interface. This command does not enable proxy ARP on the interface; see the **ip proxy-arp** command for more information on enabling proxy ARP.

The **no** variant of this command disables Local Proxy ARP to stop your device from intercepting and responding to ARP requests between hosts within a subnet. This allows the hosts to use MAC address resolution to communicate directly with one another. Local Proxy ARP is disabled by default.

Syntax

```
ip local-proxy-arp
no ip local-proxy-arp
```

Default Local proxy ARP is disabled by default

Mode Interface Configuration for a VLAN interface or a local loopback interface.

Examples To enable your device to apply Local Proxy ARP on the interface `vlan7`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan7
awplus(config-if)# ip local-proxy-arp
```

To disable your device to apply Local Proxy ARP on the interface `vlan7`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan7
awplus(config-if)# no ip local-proxy-arp
```

Related Commands

ip proxy-arp
show arp
show running-config

ip name-server

This command adds IPv4 or IPv6 DNS server addresses. The DNS client on your device sends DNS queries to IP addresses in this list when trying to resolve a host name. Host names cannot be resolved until you have added at least one server to this list. A maximum of three name servers can be added to this list.

The **no** variant of this command removes the specified DNS name-server address.

Syntax

```
ip name-server <ip-addr>
no ip name-server <ip-addr>
```

Parameter	Description
<ip-addr>	The IP address of the DNS server that is being added to the name server list. The address is entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X:X` for an IPv6 address.

Mode Global Configuration

Usage When your device is using its DHCP client for an interface, it can receive Option 6 messages from the DHCP server. This option appends the name server list with more DNS servers. See **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction** for more information about DHCP and DHCP options.

See **"DNS Relay" on page 28.11** for more information about DNS Relay to map IPv4 and IPv6 addresses to name servers to maintain a database of hostname-to-address mappings. Also see **"DNS Client" on page 28.9** for information about DNS Client configuration commands.

Examples To allow a device to send DNS queries to a DNS server with the IPv4 address `10.10.10.5`, use the commands:

```
awplus# configure terminal
awplus(config)# ip name-server 10.10.10.5
```

To enable your switch to send DNS queries to a DNS server with the IPv6 address `2001:0db8:010d::1`, use the commands:

```
awplus# configure terminal
awplus(config)# ip name-server 2001:0db8:010d::1
```

Related Commands

ip domain-list
ip domain-lookup
ip domain-name
show ip dns forwarding cache
show ip name-server

ip proxy-arp

This command enables Proxy ARP responses to ARP requests on an interface. When enabled, your device intercepts ARP broadcast packets and substitutes its own physical address for that of the remote host. By responding to the ARP request, your device ensures that subsequent packets from the local host are directed to its physical address, and it can then forward these to the remote host.

Your device responds only when it has a specific route to the address being requested, excluding the interface route that the ARP request arrived from. It ignores all other ARP requests. See the **ip local-proxy-arp** command about enabling your device to respond to other ARP messages.

The **no** variant of this command disables Proxy ARP responses on an interface. Proxy ARP is disabled by default.

Syntax

```
ip proxy-arp
no ip proxy-arp
```

Default Proxy ARP is disabled by default.

Mode Interface Configuration for a VLAN interface or a local loopback interface.

Examples To enable your device to Proxy ARP on the interface `vlan13`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan13
awplus(config-if)# ip proxy-arp
```

To disable your device to Proxy ARP on the interface `vlan13`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan13
awplus(config-if)# no ip proxy-arp
```

Related Commands

arp (IP address MAC)
ip local-proxy-arp
show arp
show running-config

ip redirects

This command enables ICMP redirects for a device.

Use the **no** variant of this command to disable the sending of ICMP redirects for a device.

Syntax

```
ip redirects
no ip redirects
```

Default ICMP redirects are disabled by default.

Mode Global Configuration

Usage ICMP redirect messages are used to notify hosts that a better route is available to a destination. ICMP redirects are used when a packet is routed into the switch on the same interface that the packet is routed out of the switch. ICMP redirects are also used when the subnet or network of the source address is on the same subnet or network as the next-hop address for a packet.

Use the **ip redirects** command to allow the sending of ICMP redirects whenever the switch receives a packet that is routed on the same interface that the packet was sent on.

Use the **no** variant of this command to disallow the sending of ICMP redirects whenever the switch receives a packet that is routed on the same interface that the packet was sent on.

Examples To enable ICMP redirects, use the following commands:

```
awplus# configure terminal
awplus(config)# ip redirects
```

To disable ICMP redirects, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip redirects
```

optimistic-nd

Use this command to enable the optimistic neighbor discovery feature for both IPv4 and IPv6.

Use the **no** variant of this command to disable the optimistic neighbor discovery feature.

Syntax

```
optimistic-nd
no optimistic-nd
```

Default The optimistic neighbor discovery feature is enabled by default.

Mode Interface Configuration for a VLAN interface.

Usage The optimistic neighbor discovery feature allows the switch, after learning an IPv4 or IPv6 neighbor, to refresh the neighbor before the neighbor is deleted from the hardware L3 switching table. The neighbor is put into the 'stale' state in the software switching table if is it not refreshed, then the 'stale' neighbors are deleted from the hardware L3 switching table.

The optimistic neighbor discovery feature enables the switch to sustain L3 traffic switching to a neighbor without interruption. Without the optimistic neighbor discovery feature enabled L3 traffic is interrupted when a neighbor is 'stale' and is then deleted from the L3 switching table.

If a neighbor receiving optimistic neighbor solicitations does not answer optimistic neighbor solicitations with neighbor advertisements, then the neighbor will be put into the 'stale' state, and subsequently deleted from both the software and the hardware L3 switching tables.

Examples To enable the optimistic neighbor discovery feature on `vlan100`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan100
awplus(config-if)# optimistic-nd
```

To disable the optimistic neighbor discovery feature on `vlan100`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan100
awplus(config-if)# no optimistic-nd
```

Validation Commands **show running-config**

ping

This command sends a query to another IPv4 host (send Echo Request messages).

Syntax

```
ping [ip] <host> [broadcast] [df-bit {yes|no}] [interval <0-128>]
   [pattern <hex-data-pattern>] [repeat {<1-2147483647>|continuous}]
   [size <36-18024>] [source <ip-addr>] [timeout <1-65535>] [tos <0-
   255>]
```

Parameter	Description
<host>	The destination IP address or hostname.
broadcast	Allow pinging of a broadcast address.
df-bit	Enable or disable the do-not-fragment bit in the IP header.
interval *<0-128>*	Specify the time interval in seconds between sending ping packets. The default is 1.
pattern *<hex-data-pattern>*	Specify the hex data pattern.
repeat	Specify the number of ping packets to send.
<1-2147483647>	Specify repeat count. The default is 5.
continuous	Continuous ping
size *<36-18024>*	The number of data bytes to send, excluding the 8 byte ICMP header. The default is 56 (64 ICMP data bytes).
source *<ip-addr>*	The IP address of a configured IP interface to use as the source in the IP header of the ping packet.
timeout *<1-65535>*	The time in seconds to wait for echo replies if the ARP entry is present, before reporting that no reply was received. If no ARP entry is present, it does not wait.
tos *<0-255>*	The value of the type of service in the IP header.

Mode User Exec and Privileged Exec

Example To ping the IP address `10.10.0.5` use the following command:

```
awplus# ping 10.10.0.5
```

show arp

Use this command to display entries in the ARP routing and forwarding table—the ARP cache contains mappings of IP addresses to physical addresses for hosts. To have a dynamic entry in the ARP cache, a host must have used the ARP protocol to access another host.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show arp [security [interface [<interface-list>]]
show arp [statistics [detail][interface [<interface-list>]]
```

Mode User Exec and Privileged Exec

Usage Running this command with no additional parameters will display all entries in the ARP routing and forwarding table.

Example To display all ARP entries in the ARP cache, use the following command:

```
awplus# show arp
```

Output **Figure 29-2: Example output from the show arp command**

```
awplus#show arp

 IP Address       MAC Address     Interface    Port         Type
192.168.10.2     0015.77ad.fad8  vlan1        port1.0.1    dynamic
192.168.20.2     0015.77ad.fa48  vlan2        port1.0.2    dynamic
192.168.1.100    00d0.6b04.2a42  vlan2        port1.0.6    static
```

Table 29-2: Parameters in the output of the show arp command

Parameter	Meaning
IP Address	IP address of the network device this entry maps to.
MAC Address	Hardware address of the network device.
Interface	Interface over which the network device is accessed.
Port	Physical port that the network device is attached to.
Type	Whether the entry is a static or dynamic entry. Static entries are added using the **arp (IP address MAC)** command. Dynamic entries are learned from ARP request/reply message exchanges.

Related Commands

arp (IP address MAC)
clear arp-cache

show debugging ip dns forwarding

Use this command to display the DNS Relay debugging status. DNS Relay debugging is set using the **debug ip dns forwarding** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging ip dns forwarding`

Mode User Exec and Privileged Exec

Example To display the DNS Relay debugging status, use the command:

```
awplus# show debugging ip dns forwarding
```

Output **Figure 29-3: Example output from the show debugging ip dns forwarding command**

```
awplus#show debugging ip dns forwarding
DNS Relay debugging status:
  debugging is on
```

Related Commands **debug ip dns forwarding**

show debugging ip packet

Use this command to show the IP interface debugging status. IP interface debugging is set using the **debug ip packet interface** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging ip packet`

Mode User Exec and Privileged Exec

Example To display the IP interface debugging status when the terminal monitor off, use the command:

```
awplus# terminal no monitor
awplus# show debug ip packet
```

Output **Figure 29-4: Example output from the show debugging ip packet command with terminal monitor off**

```
awplus#terminal no monitor
awplus#show debug ip packet
IP debugging status:
interface all tcp (stopped)
interface vlan1 arp verbose (stopped)
```

Example To display the IP interface debugging status when the terminal monitor is on, use the command:

```
awplus# terminal monitor
awplus# show debug ip packet
```

Output **Figure 29-5: Example output from the show debugging ip packet command with terminal monitor on**

```
awplus#terminal monitor
awplus#show debug ip packet
IP debugging status:
interface all tcp (running)
interface vlan1 arp verbose (running)
```

Related Commands **debug ip packet interface**
terminal monitor

show hosts

This command shows the default domain, domain list, and name servers configured on your device.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show hosts
```

Mode

User Exec and Privileged Exec

Example

To display the default domain, use the command:

```
awplus# show hosts
```

Output

Figure 29-6: Example output from the show hosts command

```
awplus#show hosts
Default domain is mycompany.com
Domain list: company.com
Name/address lookup uses domain service
Name servers are 10.10.0.2 10.10.0.88
```

Related Commands

ip domain-list
ip domain-lookup
ip domain-name
ip name-server

show ip dns forwarding

Use this command to display the DNS Relay status.

Syntax `show ip dns forwarding`

Mode User Exec and Privileged Exec

Examples To display the DNS Relay status, use the command:

```
awplus# show ip dns forwarding
```

Output **Figure 29-7: Example output from the show ip dns forwarding command**

```
awplus#show ip dns forwarding
Max-Retry       : 2
Timeout         : 3 second(s)
Dead-Time       : 3600 second(s)
Source-Interface: not specified
DNS Cache       : disabled
```

Related Commands ip dns forwarding

show ip dns forwarding cache

Use this command to display the DNS Relay name resolver cache.

Syntax `show ip dns forwarding cache`

Mode User Exec and Privileged Exec

Example To display the DNS Relay name resolver cache, use the command:

```
awplus# show ip dns forwarding cache
```

Output **Figure 29-8: Example output from the show ip dns forwarding cache command**

```
awplus#show ip dns forwarding cache
Host                                    Address                 Expires Flags
www.example.com                         172.16.1.1.                 180
mail.example.com                        www.example.com             180 CNAME
www.example.com                         172.16.1.1.                 180 REVERSE
mail.example.com                        172.16.1.5.                 180
```

Example To display the DNS Relay name resolver cache with output for VRF lite instance RED, use the command:

```
awplus# show ip dns vrf RED forwarding cache
```

Related Commands **ip dns forwarding cache**
ip name-server

show ip dns forwarding server

Use this command to display the status of DNS forwarding name servers.

Syntax

```
show ip dns forwarding server
```

Parameter	Description
`forwarding server`	The DNS forwarding name server.

Mode User Exec and Privileged Exec

Examples To display the status of DNS Relay name servers, use the command:

```
awplus# show ip dns forwarding server
```

Output **Figure 29-9: Example output from the show ip dns forwarding server command**

```
awplus#show ip dns forwarding server
Servers           Forwards      Fails    Dead-Time
172.16.1.1              12          0       active
172.16.1.2               6          3         3900
```

Related Commands **ip dns forwarding**
ip dns forwarding dead-time

show ip domain-list

This command shows the domains configured in the domain list. The DNS client uses the domains in this list to append incomplete hostnames when sending a DNS inquiry to a DNS server.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip domain-list`

Mode User Exec and Privileged Exec

Example To display the list of domains in the domain list, use the command:

```
awplus# show ip domain-list
```

Output **Figure 29-10: Example output from the show ip domain-list command**

```
awplus#show ip domain-list
alliedtelesis.com
mycompany.com
```

Related Commands **ip domain-list**
ip domain-lookup

show ip domain-name

This command shows the default domain configured on your device. When there are no entries in the DNS list, the DNS client appends this domain to incomplete hostnames when sending a DNS inquiry to a DNS server.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip domain-name`

Mode User Exec and Privileged Exec

Example To display the default domain configured on your device, use the command:

```
awplus# show ip domain-name
```

Output **Figure 29-11: Example output from the show ip domain-name command**

```
awplus#show ip domain-name
alliedtelesis.com
```

Related Commands **ip domain-name**
ip domain-lookup

show ip interface

Use this command to display information about interfaces and the IP addresses assigned to them. To display information about a specific interface, specify the interface name with the command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip interface [<interface-list>] [brief]
```

Parameter	Description
<interface-list>	The interfaces to display information about. An interface-list can be: ■ an interface, e.g. `vlan2` ■ a continuous range of interfaces separated by a hyphen, e.g. `vlan2-8` or `vlan2-vlan5` ■ a comma-separated list of interfaces or interface ranges, e.g. `vlan2,vlan5,vlan8-10` The specified interfaces must exist.

Mode User Exec and Privileged Exec

Examples To show brief information for the assigned IP address for interface port1.0.2 use the command:

```
awplus# show ip interface port1.0.2 brief
```

To show the IP addresses assigned to vlan2 and vlan3, use the command:

```
awplus# show ip interface vlan2-3 brief
```

Output **Figure 29-12: Example output from the show ip interface brief command**

```
Interface              IP-Address      Status          Protocol
port1.0.2              unassigned      admin up        down
vlan1                  192.168.1.1     admin up        running
vlan2                  192.168.2.1     admin up        running
vlan3                  192.168.3.1     admin up        running
vlan8                  unassigned      admin up        down
```

show ip name-server

This command displays a list of IPv4 and IPv6 DNS server addresses that your switch will send DNS requests to. This is a static list configured using the **ip name-server** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip name-server
```

Mode

User Exec and Privileged Exec

Example

To display the list of DNS servers that your device sends DNS requests to, use the command:

```
awplus# show ip name-server
```

Output

Figure 29-13: Example output from the show ip name-server command

```
awplus# show ip name-server
 10.10.0.123
 10.10.0.124
 2001:0db8:010d::1
```

Related Commands

ip domain-lookup
ip name-server

show ip sockets

Use this command to display information about the IP or TCP sockets that are present on the switch. It includes TCP, UDP listen sockets, displaying associated IP address and port.

The information displayed for established TCP sessions includes the remote IP address, port, and session state. Raw IP protocol listen socket information is also displayed for protocols such as VRRP and ICMP6, which are configured to receive IP packets with the associated protocol number.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip sockets`

Mode Privileged Exec

Usage Use this command to verify that the socket being used is opening correctly. If there is a local and remote endpoint, a connection is established with the ports indicated.

Note that this command does not display sockets that are used internally for exchanging data between the various processes that exist on the device and are involved in its operation and management. It only displays sockets that are present for the purposes of communicating with other external devices.

Example To display ip sockets currently present on the device, use the command:

```
awplus# show ip sockets
```

Output **Figure 29-14: Example output from the show ip sockets command**

```
Socket information

Not showing 40 local connections
Not showing 7 local listening ports

Typ Local Address            Remote Address          State
tcp 0.0.0.0:111              0.0.0.0:*               LISTEN
tcp 0.0.0.0:80               0.0.0.0:*               LISTEN
tcp 0.0.0.0:23               0.0.0.0:*               LISTEN
tcp 0.0.0.0:443              0.0.0.0:*               LISTEN
tcp 0.0.0.0:4743             0.0.0.0:*               LISTEN
tcp 0.0.0.0:873              0.0.0.0:*               LISTEN
tcp :::23                    :::*                    LISTEN
udp 0.0.0.0:111              0.0.0.0:*
udp 226.94.1.1:5405          0.0.0.0:*
udp 0.0.0.0:161              0.0.0.0:*
udp :::161                   :::*
raw 0.0.0.0:112              0.0.0.0:*               112
raw :::58                    :::*                    58
raw :::112                   :::*                    112
```

Table 29-3: Parameters in the output of the show ip sockets command

Parameter	Description
`Not showing <number> local connections`	This field refers to established sessions between processes internal to the device, that are used in its operation and management. These sessions are not displayed as they are not useful to the user. <number> is some positive integer.
`Not showing <number> local listening ports`	This field refers to listening sockets belonging to processes internal to the device, that are used in its operation and management. They are not available to receive data from other devices. These sessions are not displayed as they are not useful to the user. <number> is some positive integer.
`Typ`	This column displays the type of the socket. Possible values for this column are: `tcp` : IP Protocol 6 `udp` : IP Protocol 17 `raw` : Indicates that socket is for a non port-orientated protocol (i.e. a protocol other than TCP or UDP) where all packets of a specified IP protocol type are accepted. For raw socket entries the protocol type is indicated in subsequent columns.
`Local Address`	For TCP and UDP listening sockets this shows the destination IP address (either IPv4 or IPv6) and destination TCP or UDP port number for which the socket will receive packets. The address and port are separated by ':'. If the socket will accept packets addressed to any of the switch's IP addresses, the IP address will be 0.0.0.0 for IPv4 or :: for IPv6. For active TCP sessions the IP address will display which of the switches addresses the session was established with. For raw sockets this displays the IP address and IP protocol for which the socket will accept IP packets. The address and protocol are separated by ':'. If the socket will accept packets addressed to any of the switch's IP addresses, the IP address will be 0.0.0.0 for IPv4 and :: for IPv6. IP Protocol assignments are described at: **www.iana.org/assignments/protocol-numbers**
`Remote Address`	For TCP and UDP listening sockets this shows the source IP address (either IPv4 or IPv6) and source TCP or UDP port number for which the socket will accept packets. The address and port are separated by ':'. If the socket will accept packets addressed from any IP address, the IP address will be 0.0.0.0 for IPv4 or :: for IPv6. This is the usual case for a listening socket. Normally for a listen socket any source port will be accepted. This is indicated by ". For active TCP sessions the IP address will display the remote address and port the session was established with. For raw sockets the entry in this column will be 0.0.0.0: or ::: for IPv4 and IPv6, respectively.

`State`	This column shows the state of the socket. For TCP sockets this shows the state of the TCP state machine. For UDP sockets this column is blank. For raw sockets it contains the IP protocol number. The possible TCP states are:
	`LISTEN`
	`SYN-SENT`
	`SYN-RECEIVED`
	`ESTABLISHED`
	`FIN-WAIT-1`
	`FIN-WAIT-2`
	`CLOSE-WAIT`
	`CLOSING`
	`LAST-ACK`
	`TIME-WAIT`
	`CLOSED`
	RFC793 contains the TCP state machine diagram with Section 3.2 describing each of the states.

show ip traffic

Use this command to display statistics regarding IP traffic sent and received by all interfaces on the switch, showing totals for IP and IPv6 and then broken down into sub-categories such as TCP, UDP, ICMP and their IPv6 equivalents when appropriate.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip traffic`

Mode Privileged Exec

Example To display IP traffic statistics, use the command:

```
awplus# show ip traffic
```

Output **Figure 29-15: Example output from the show ip traffic command**

```
IP:
     261998 packets received
     261998 delivered
     261998 sent
     69721 multicast packets received
     69721 multicast packets sent
     23202841 bytes received
     23202841 bytes sent
     7669296 multicast bytes received
     7669296 multicast bytes sent
IPv6:
     28 packets discarded on transmit due to no route
ICMP6:
UDP6:
UDPLite6:
TCP:
     0 remote connections established
     40 local connections established
     7 remote listening ports
     7 local listening ports
     261 active connection openings
     247 passive connection openings
     14 connection attempts failed
     122535 segments received
     122535 segments transmitted
     14 resets transmitted
     227 TCP sockets finished time wait in fast timer
     155 delayed acks sent
     21187 headers predicted
     736 pure ACKs
     80497 pure ACKs predicted
UDP:
     139468 datagrams received
     139468 datagrams sent
UDPLite:
```

Table 29-4: Parameters in the output of the show ip traffic command

Parameter	Description
IPv4	IPv4 counters
IPv6	IPv6 counters

Table 29-4: Parameters in the output of the show ip traffic command (cont.)

Parameter	Description
`received packets with no route`	Received packets with no route
`truncated packets received`	Truncated packets received
`multicast packets received`	Multicast packets received
`multicast packets sent`	Multicast packets sent
`broadcast packets received`	Broadcast packets received
`broadcast packets sent`	Broadcast packets sent
`bytes received`	Bytes received
`bytes sent`	Bytes sent
`multicast bytes received`	Multicast bytes received
`multicast bytes sent`	Multicast bytes sent
`broadcast bytes received`	Broadcast bytes received
`broadcast bytes sent`	Broadcast bytes sent
`packets received`	Packets received
`packets received with invalid headers`	Packets received with invalid headers
`oversize packets received`	Oversize packets received
`packets received with no route`	Packets received with no route
`packets received with invalid address`	Packets received with invalid address
`packets received with unknown protocol`	Packets received with unknown protocol
`truncated packets received`	Truncated packets received
`received packets discarded`	Received packets discarded
`received packets delivered`	Received packets delivered
`forwarded packets transmitted`	Forwarded packets transmitted
`packets transmitted`	Packets transmitted
`packets discarded on transmit`	Packets discarded on transmit
`packets discarded on transmit due to no route`	Packets discarded on transmit due to no route
`fragment reassembly timeouts`	Fragment reassembly timeouts
`fragment reassembly required`	Fragment reassembly required
`fragment reassembly OK`	Fragment reassembly OK
`fragment reassembly failures`	Fragment reassembly failures
`fragmentations succeeded`	Fragmentations succeeded

Table 29-4: Parameters in the output of the show ip traffic command (cont.)

Parameter	Description
`fragmentations failed`	Fragmentations failed
`fragments created`	Fragments created
`ICMP6`	ICMPv6 counters
`messages received`	Messages received
`errors received`	Errors received
`messages sent`	Messages sent
`TCP`	TCP counters
`remote connections established`	Remote connections established
`local connections established`	Local connections established
`remote listening ports`	Remote listening ports
`local listening ports`	Local listening ports
`active connection openings`	Active connection openings
`passive connection openings`	Passive connection openings
`connection attempts failed`	Connection attempts failed
`connection resets received`	Connection resets received
`segments received`	Segments received
`segments transmitted`	Segments transmitted
`retransmits`	Retransmits
`bad segments received`	Bad segments received
`resets transmitted`	Resets transmitted
`datagrams received`	Datagrams received
`received for unknown port`	Received for unknown port
`datagrams sent`	Datagrams sent
`syncookies sent`	Syncookies sent
`syncookies received`	Syncookies received
`syncookies failed`	Syncookies failed
`embryonic resets`	Embryonic resets
`sockets pruned`	Sockets pruned
`ICMPs out of window`	ICMPs out of window
`ICMPs dropped due to lock`	ICMPs dropped due to lock
`ARPs filtered`	ARPs filtered

Table 29-4: Parameters in the output of the show ip traffic command (cont.)

Parameter	Description
`TCP sockets finished time wait in fast timer`	TCP sockets finished time wait in fast timer
`time wait sockets recycled by time stamp`	Time wait sockets recycled by time stamp
`time wait sockets killed`	Time wait sockets killed
`delayed acks sent`	Delayed acks sent delayed acks further delayed because of locked socket
`delayed acks lost`	Delayed acks lost
`listening socket overflows`	Listening socket overflows
`listening socket drops`	Listening socket drops
`headers predicted`	Headers predicted
`pure ACKs`	Pure ACKs
`pure ACKs predicted`	Pure ACKs predicted
`losses recovered by TCP Reno`	Losses recovered by TCP Reno
`losses recovered by SACK`	Losses recovered by SACK
`SACKs renegged`	SACKs renegged
`detected reordering by FACK`	Detected reordering by FACK
`detected reordering by SACK`	Detected reordering by SACK
`detected reordering by TCP Reno`	Detected reordering by TCP Reno
`detected reordering by sequence`	Detected reordering by sequence
`full undos`	Full undos
`partial undos`	Partial undos
`SACK undos`	SACK undos
`loss undos`	Loss undos
`segments lost`	Segments lost
`lost retransmits`	Lost retransmits
`TCP Reno failures`	TCP Reno failures
`SACK failures`	SACK failures
`loss failures`	Loss failures
`fast retransmits`	Fast retransmits
`forward retransmits`	Forward retransmits
`retransmits in slow start`	Retransmits in slow start

Table 29-4: Parameters in the output of the show ip traffic command (cont.)

Parameter	Description
`timeouts`	Timeouts
`TCP Reno recovery failures`	TCP Reno recovery failures
`SACK recovery failures`	SACK recovery failures
`collapsed segments received`	Collapsed segments received
`DSACKs sent for old packets`	DSACKs sent for old packets
`DSACKs sent for out of order segments`	DSACKs sent for out of order segments
`DSACKs received`	DSACKs received
`DSACKs received for out of order segments`	DSACKs received for out of order segments
`connections reset due to unexpected SYN`	Connections reset due to unexpected SYN
`connections reset due to unexpected data`	Connections reset due to unexpected data
`connections reset due to early user close`	Connections reset due to early user close
`connections aborted due to lack of memory`	Connections aborted due to lack of memory
`connections aborted due to timeout`	Connections aborted due to timeout
`connections aborted due to lingering`	Connections aborted due to lingering
`connection aborts due to connection failure`	Connection aborts due to connection failure
`TCP memory pressure events`	TCP memory pressure events
`SACKs discarded`	SACKs discarded
`Old DSACKs ignored`	Old DSACKs ignored
`DSACKs ignored without undo`	DSACKs ignored without undo
`Spurious RTOs`	Spurious RTOs
`TCP MD5 Not Found`	TCP MD5 Not Found
`TCP MD5 Unexpected`	TCP MD5 Unexpected
`TCP SACKs shifted`	TCP SACKs shifted
`TCP SACKs merged`	TCP SACKs merged
`TCP SACK shift fallback`	TCP SACK shift fallback
`UDP`	UDP Counters
`UDPLite`	UDPLite Counters

Table 29-4: Parameters in the output of the show ip traffic command (cont.)

Parameter	Description
`UDP6`	UDPv6 Counters
`UDPLite6`	UDPLitev6 Counters
`datagrams received`	Datagrams received
`datagrams received for unknown port`	Datagrams received for unknown port
`datagram receive errors`	Datagram receive errors
`datagrams transmitted`	Datagrams transmitted
`datagrams received`	Datagrams received
`datagrams received for unknown port`	Datagrams received for unknown port
`datagram receive errors`	Datagram receive errors
`datagrams transmitted`	Datagrams transmitted

tcpdump

Use this command to start a tcpdump, which gives the same output as the Unix-like **tcpdump** command to display TCP/IP traffic. Press `<ctrl> + c` to stop a running tcpdump.

Syntax `tcpdump <line>`

Parameter	Description
<line>	Specify the dump options. For more information on the options for this placeholder see URL **www.tcpdump.org/tcpdump_man.html**

Mode Privileged Exec

Example To start a tcpdump running to capture IP packets, enter the command:

```
awplus# tcpdump ip
```

Output **Figure 29-16: Example output from the tcpdump command**

```
03:40:33.221337 IP 192.168.1.1 > 224.0.0.13: PIMv2, Hello,
length: 34
1 packets captured
2 packets received by filter
0 packets dropped by kernel
```

Related Commands **debug ip packet interface**

traceroute

Use this command to trace the route to the specified IPv4 host.

Syntax `traceroute {<ip-addr>|<hostname>}`

Parameter	Description
<ip-addr>	The destination IPv4 address. The IPv4 address uses the format A.B.C.D.
<hostname>	The destination hostname.

Mode User Exec and Privileged Exec

Example

```
awplus# traceroute 10.10.0.5
```

This command applies the functionality of the **no debug ip packet interface** command on page 29.11.

Chapter 30: IPv6 Introduction

Introduction 30.2
Overview 30.2
IPv6 Addresses and Prefixes 30.3
Address Types 30.3
IPv6 Headers 30.5
The Internet Control Message Protocol (ICMPv6) 30.7
IPv6 Routing 30.10
Integration of IPv4 and IPv6 ..... 30.10
IPv6 on your Switch 30.11
Enabling IPv6 ..... 30.11
IPv6 Stateless Address Autoconfiguration (SLAAC) 30.11
IPv6 EUI-64 Addressing 30.11
IPv6 Link-local Addresses 30.12
IPv6 RA Guard 30.13
RA Guard Introduction 30.13
Enabling IPv6 RA Guard 30.13

Introduction

This chapter describes the main features of IPv6, the switch's implementation of IPv6 and how to configure and operate IPv6 on the switch.

This chapter describes the following IPv6 features:

- linking together networks that run IPv6.
- allowing address autoconfiguration of hosts connected to the switch.

Overview

IPv6 is the next generation of the Internet Protocol (IP). It has primarily been developed to solve the problem of the eventual exhaustion of the IPv4 address space, but also offers other enhancements. IPv6 addresses are 16 bytes long, in contrast to IPv4's 4 byte addresses. Other features of IPv6 include:

- Address structure improvements:
  - « globally unique addresses with more levels of addressing hierarchy to reduce the size of routing tables
  - « autoconfiguration of addresses by hosts
  - « improved scalability of multicast routing by adding a "scope" field to multicast addresses
  - « a new type of address, the "anycast address", which sends packets to any one of a group of devices
- Removes the need for packet fragmentation en-route, by dynamic determination of the largest packet size that is supported by every link in the path. A link's MTU (Maximum Transmission Unit) must be at least 1280 bytes, compared with 576 bytes for IPv4.
- Includes a Traffic Class that allow packets to be labeled with an appropriate priority. If the network becomes congested, the lowest priority packets are dropped.
- Includes Flow labels that indicate to intermediate switches and routers that packets are part of a flow, and that a particular flow requires a particular type of service. This feature enables, for example, real-time processing of data streams. It also increases routing speed because the forwarding router or switch needs only to check the flow label, not the rest of the header. The handling indicated by the flow label can be done by the IPv6 Hop-by-Hop header, or by a separate protocol such as RSVP.
- Mandatory authentication and data integrity protocols through IPsec. IPsec is optional in IPv4.

IPv6 Addresses and Prefixes

IPv6 addresses have a hexadecimal format that is made up of eight pairs of octets separated by colons. An example of a valid address is **2001:0db8:0000:0000:0260:0000:97ff:64aa**. In the interests of brevity, addresses can be abbreviated in two ways:

- Leading zeros can be omitted, so this address can be written as **2001:db8:0:0:260:0:97ff:64aa**.
- Consecutive zeros can be replaced with a double colon, so this address can be written as **2001:db8::260:0:97ff:64a**. Note that a double colon can replace any number of consecutive zeros, but an address can contain only one double colon.

Like IPv4 addresses, a proportion of the leftmost bits of the IPv6 address can be used to indicate the subnet, rather than a single node. This part of the address is called the *prefix*. Prefixes provide the equivalent functionality to a subnet mask in IPv4, allowing a subnet to be addressed, rather than a single node. If a prefix is specified, the IPv6 address is followed by a slash and the number of bits that represent the prefix. For example, **2001::/16** indicates that the first 16 bits (**2001**) of the address **2001:0:0:0:0:0:0:0** represent the prefix. **Note that your switch does not allow prefixes greater than 64 bits**.

Like IPv4 addresses, IPv6 addresses are attached to interfaces.

Address Types

IPv6 supports the following address types:

- Unicast
- Multicast
- Anycast

Unicast addresses

A unicast address is attached to a single interface and delivers packets only to that interface. The following special addresses have been defined:

- IPv4-compatible and IPv4-mapped addresses. IPv4-compatible addresses are used to tunnel IPv6 packets across an IPv4 network. IPv4-mapped addresses are used by an IPv6 host to communicate with an IPv4 host. The IPv6 host addresses the packet to the mapped address.
- Link-local addresses can be used on the local network on which the interface is attached. The link-local prefix is **fe80::/10**. Different interfaces on a device may have the same link-local address. The switch will automatically generate a link-local address for all interfaces that are using IPv6. Commands entered to configure link-local addresses that match any automatically generated link-local addresses by the switch will not be executed.

  Enter the **show ipv6 interface** command to display automatically generated link-local addresses not shown in the **running-config**. Automatically generated link-local addresses contain the last six hexadecimal numbers of the MAC address for a given interface.
- The Loopback address, consisting of **::1**, which is the equivalent of the IPv4 loopback address and allows a host to send packets to itself.
- The Unspecified address, consisting of **::**, which is the equivalent of the IPv4 unspecified address and is used as a source address by hosts during the autoconfiguration process.

Multicast addresses

IPv6 multicast addresses provide the equivalent functionality to broadcast addresses in IPv4. Broadcast addresses are not supported in IPv6. A multicast address identifies a group of interfaces, and packets are sent to all interfaces in that group.

Among the special addresses that have been defined are addresses that allow multicasting to:

- All interfaces on a particular host (**ff01::1**)
- All nodes on a local network (**ff01::2**)
- All routers on the local link (**ff02::2**)
- All routers on the local site (**ff05::2**).

Anycast addresses

An *anycast* address is a unicast address that is attached to more than one interface. If a packet is sent to an anycast address it is delivered to the nearest interface with that address, with the definition of "nearest" depending on the protocol used for routing. If the protocol is RIPv6, the nearest interface is the one that is the shortest number of hops away.

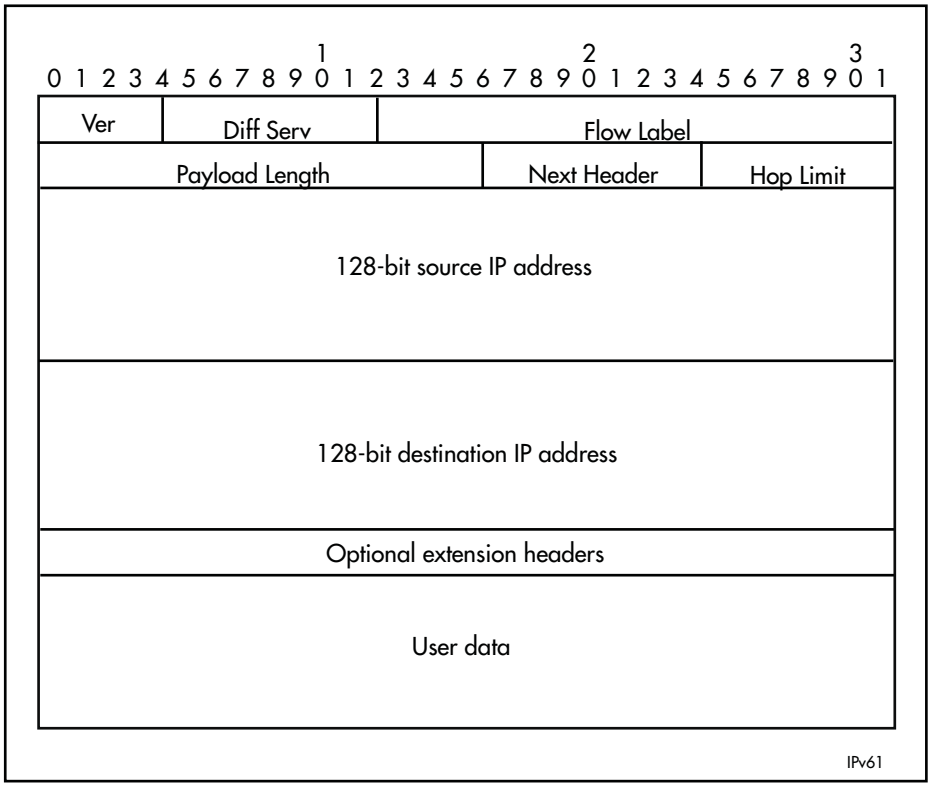
Anycast addresses can be assigned to routers only, and packets cannot originate from an anycast address. A router must be configured to know that it is using an anycast address because the address format cannot be distinguished from that of a unicast address.

Only one anycast address has been predefined: the subnet-router address. The subnet-router address sends messages to the nearest router on a subnet and consists of the subnet's prefix followed by zeros.

IPv6 Headers

The basic unit of data sent through an internet is called a *packet* in IPv6. A packet consists of a *header* followed by the *data*. The following figure shows the IPv6 packet.

Figure 30-1: IPv6 packet



.

Table 30-1: IPv6 packet - Field Description

Field	Function
Ver	Version of the IP protocol that created the packet. For IPv6, this field has a value of 6.
Differentiated Services	8-bit value that contains the 6-bit DSCP and is used to prioritize traffic as part of a Quality of Service system. For more information, see **"Differentiated Services Architecture" on page 62.4**. Additional information can be found in RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*.
Flow Label	20-bit value that indicates the data flow to which this packet belongs.This flow may be handled in a particular way.
Payload Length	Length of the user data portion of the packet. If the data payload is larger than 64 kB, the length is given in the optional "Jumbo Payload" header and the Payload Length header is given a value of zero.

Table 30-1: IPv6 packet - Field Description(cont.)

Field	Function
Next Header	Number that indicates the type of header that immediately follows the basic IP header. This header type may be an optional IPv6 extension header, a relevant IPv4 option header, or another protocol, such as TCP or ICMPv6. The IPv6 extension header values are: **0** (Hop-by-Hop Options Header) **43** (IPv6 Routing Header) **44** (IPv6 Fragment Header) **50** (Encapsulating Security Payload) **51** (IPv6 Authentication Header) **59** (No Next Header) **60** (Destination Options Header)
Hop Limit	Field that is the equivalent of the IPv4 Time To Live field, measured in hops.
Source IP address	128-bit IPv6 address of the sender.
Destination IP address	128-bit IPv6 address of the recipient.
Optional extension headers	Headers for less-frequently used information.
User data	Payload.

Basic IPv6 header structure

The headers contain information necessary to move the packet across the Internet. They must be able to cope with missing and duplicated packets as well as possible fragmentation (and reassembly) of the original packet.

IPv6 headers are twice as long as IPv4 headers (40 bytes instead of 20 bytes) and contain four times the address space size (128 bits instead of 32 bits).

They no longer contain the header length, identification, flags, fragment offset, and header checksum fields. Some of these options are placed in extension headers. The Time To Live field is replaced with a hop limit, and the IPv4 Type of Service field is replaced with a Differentiated Services field. The Differentiated Services field contains the DSCP bits, used in a Quality of Service (QoS) regime. The following table explains IPv4 header fields that changed in IPv6.

Changed Field	Description
Type of Service	The type of service that a connection should receive is indicated in IPv6 by the Flow Label field in the IPv6 header.
Fragmentation information (the Identification field, the Flags field and the Fragment Offset field)	In most cases fragmentation does not occur in IPv6. If it does, packets are fragmented at their source and not en route. Therefore, the fragmentation information is contained in an extension header to reduce the size of the basic IPv6 header.
Header Checksum	This option has not been provided in IPv6. This is because transport protocols implement checksums and because of the availability of the IPsec authentication header (AH) in IPv6.
Options	Extension headers handle all the optional values associated with IPv6 packets. The biggest advantage of this scheme is that the size of the basic IP header is a constant.

Extension headers

IPv6 implements many of the less commonly used fields in the IPv4 header (or their equivalents) as extension headers, which are placed after the basic IPv6 header. The length of each header must be a multiple of 8 bytes.

The first extension header is identified by the Next Header field in the basic IPv6 header. Any subsequent extension headers are identified by an 8-bit "Next Header" value at the beginning of the preceding extension header.

IPv6 nodes that originate packets are required to place extension headers in a specific order:

1. The basic IPv6 header. This must come immediately before the extension headers.
2. The Hop-by-Hop header. This specifies options that must be examined by every node in the routing path.
3. A Destination Options header. This is used to specify options to be processed by the first destination or final destination. The destination options header is the only extension header that may be present more than once in the IPv6 packet.
4. The Routing header. This enables a static path to be specified for the packet, if the dynamically-determined path is undesirable.
5. The Fragment header. This indicates that the source node has fragmented the packet, and contains information about the fragmentation.
6. The Authentication header (AH). This verifies the integrity of the packet and its headers.
7. The Encapsulating Security Payload (ESP) header. This encrypts a packet and verifies the integrity of its contents.
8. The Upper Layer Protocol header. This indicates which protocol a higher layer (such as the transport layer) is to process the packet with (for example, TCP).

The Internet Control Message Protocol (ICMPv6)

The Internet Control Message Protocol, ICMPv6, provides a mechanism for error reporting and route discovery and diagnostics. It also conveys information about multicast group membership, a function that is carried out by the Internet Group Management Protocol (IGMP) in IPv4, and performs address resolution, which the Address Resolution Protocol (ARP) performs in IPv4.

Significant aspects of ICMPv6 include neighbor discovery, which enables one device in a network to find out about other nearby devices; and stateless address autoconfiguration, which allows a device to dynamically determine its own IPv6 address.

ICMPv6 is also used to support the Ping v6 (*Packet Internet Groper*) and Trace route v6 functions that are used to verify the connections between networks and network devices. Ping is used to test the connectivity between two network devices to determine whether each network device can "see" the other device. Trace route is used to discover the route used to pass packets between two systems running the IP protocol.

Both of these functions operate almost identically in IPv4 and IPv6. For more information, see **"Ping" on page 28.14**.

Neighbor discovery

Neighbor discovery is an ICMPv6 function that enables a router or a host to identify other devices on its links. This information is then used in address autoconfiguration, to redirect a node to use a more appropriate router if necessary, and to maintain reachability information with its neighbors.

The IPv6 Neighbor Discovery protocol is similar to a combination of the IPv4 protocols ARP, ICMP Router Discovery and ICMP Redirect.

The following table describes packet types involved with neighbor discovery.

Packet Type	Description
router solicitation	Packet in which a host sends out a request for routers to generate advertisements.
router advertisement	Allows routers to advertise their presence and other network parameters. A router sends an advertisement packet in response to a solicitation packet from a host.
neighbor solicitation	Packet in which a node sends a packet to determine the link layer address of a neighbor or to verify that a neighbor is still active.
neighbor advertisement	A response to a neighbor solicitation packet. These packets are also used to notify neighbors of link layer address changes.
redirect	Informs hosts of a better first hop.

To comply with Section 6.2.1 of RFC 2461, *IPv6 Neighbor Discovery*, the router does not generate router advertisements by default.

The following table explains packet types and services.

Packet Type	Description
address resolution	A method for carrying out address autoconfiguration, and is achieved using the Neighbor Solicitation Message and the Neighbor Advertisement Message.
router and prefix discovery	On connection to a link, a node needs to know the address of a router that the node can use to reach the rest of the world. The node also needs to know the prefix (or prefixes) that define the range of IP addresses on its link that it can reach without going through a router. Routers use ICMP to convey this information to hosts, by means of router advertisements. The message may have an option attached (the *source link address* option), which enables the receiving node to respond directly to the router, without performing a neighbor solicitation.
immediate information	The configuration of a router includes a defined frequency at which unsolicited advertisements are sent. If a node wants to obtain information about the nearest router immediately, rather than waiting for the next unsolicited advertisement, the node can send a router solicitation message. Each router that receives the solicitation message sends a router advertisement specifically to the node that sent the solicitation.

Packet Type	Description
redirection	If a node is aware of more than one router that it can use to connect to wider networks, the router to which it sends packets by default does not always represent the most desirable route. ICMPv6 uses the redirect packet to communicate a more effective path to the node.
Neighbor Unreachability Detection (NUD)	A node may issue solicitation requests to determine whether a path is still viable, or may listen in on acknowledgement packets of higher layer protocols, such as TCP. If the node determines that a path is no longer viable, it attempts to establish a new link to the neighbor, or to re-establish the previous link. NUD can be used between any two devices in the network, independent of whether the devices are acting as hosts or routers.

Stateless address autoconfiguration

Stateless address autoconfiguration allows an IPv6-aware device to be plugged into a network without manual configuration with an IP address. This plug and play functionality results in networks that are easier to set up and modify, and simplifies the process of shifting to use a new Internet Service Provider (ISP).

Stateless address autoconfiguration is achieved in a series of steps. Routers and hosts perform the first three steps, which autoconfigure a link-local address. A global address is autoconfigured in the last three steps, which only hosts perform.

On the router or host

1. During system start-up, the node begins autoconfiguration by generating a link-local address for the interface. A link-local address is formed by adding the interface ID to the link-local prefix **fe80::/10** (reference RFC 3513).

Note

Different interfaces on a device may have the same link-local address. The switch will automatically generate a link-local address for all interfaces that are using IPv6. Commands entered to configure link-local addresses that match any automatically generated link-local addresses by the switch will not be executed. Enter the show ipv6 interface command to display automatically generated link-local addresses not shown in the running-config. Automatically generated link-local addresses contain the last six hexadecimal numbers of the MAC address for a given interface.

2. The node then transmits a neighbor solicitation message to this address. If the address is already in use, the node that the address belongs to replies with a neighbor advertisement message. The autoconfiguration process stops and manual configuration of the node is then required.

3. If no neighbor advertisement is received, the node concludes that the address is available and assigns it to the chosen interface.

On the host

1. The node then sends one or more router solicitations to detect if any routers are present. Any routers present responds with a router advertisement.

   If no router advertisement is received, the node tries to use DHCP to obtain an address and other configuration information. If no DHCP server responds, the node continues using the link-level address

   If a router advertisement is received, this message informs the node how to proceed with the auto configuration process. The prefix from the router advertisement, if received, is added to the link-level address to form the global unicast IP address.

2. This address is then assigned to the network interface.

   If routers are present, the node continues to receive router advertisements. The node updates its configuration when there are changes in the router advertisements.

IPv6 Routing

Routing in IPv6 is almost identical to IPv4 routing under CIDR, except that the addresses are 128-bit IPv6 addresses instead of 32-bit IPv4 addresses.

Routing Information Protocol (RIPv6)

RIP is a simple distance vector protocol that defines networks based on how many hops they are from the router. When a network is more than 15 hops away (one hop is one link), it is not included in the routing table.

RIPv6, also referred to as RIPng (for "next generation") is similar to RIPv2. Extensions to RIPv2 to support IPv6 are:

- the address field of a routing entry is expanded to 128 bits to allow IPv6 prefixes
- the 32-bit RIPv2 subnet mask field is replaced by an 8-bit prefix length field
- authentication is removed in RIPv6
- the size of a routing packet is no longer arbitrarily limited
- RIPv6 specifies the next hop instead of simply allowing the recipient of the update to set the next hop to the sender of the update.

In RIPv6, each router uses a routing table to keep track of every destination that is reachable throughout the system. Each entry in the routing table contains:

- the IPv6 prefix of the destination
- a metric, which represents the total cost of getting a packet from the router to that destination
- the IPv6 address of the next router along the path to the destination
- a flag to indicate that information about the route has changed recently
- various timers associated with the route.

Integration of IPv4 and IPv6

IPv6 has been designed in such a way that a smooth transition from IPv4 is possible. The most effective way to ensure this is to use a *dual IP stack*. A node configured as a dual stack system has both a 128-bit IPv6 address and a 32-bit IPv4 address, and so can communicate with nodes running IPv4 and those running IPv6.

Another aspect of the transition is to *tunnel* IPv6 packets through an IPv4 network. IPv6 packets are tunneled simply by encapsulating the IPv6 packet within an IPv4 datagram, and identifying that this datagram is an encapsulated IPv6 packet by giving the datagram a protocol value of 41.

IPv6 on your Switch

This section describes the switch's support for IPv6, and how to configure IPv6 on the switch.

Enabling IPv6

The switch's implementation of IPv6 is disabled by default. To enable IPv6 forwarding, use the **ipv6 forwarding** command on page 31.8.

To display information about IPv6 settings, use the **show ipv6 interface brief** command on page 31.25.

Because the switch implements IPv6 as a dual stack, implementing IPv6 does not affect IPv4 functionality.

IPv6 Stateless Address Autoconfiguration (SLAAC)

The switch's implementation of IPv6 supports SLAAC on an interface. To enable IPv6 SLAAC on an interface, use the **ipv6 address autoconfig** command on page 31.5. SLAAC automatically applies the MAC address of the interface to an IPv6 address for the interface specified.

ipv6 address autoconfig enables automatic configuration of IPv6 addresses on an interface using stateless autoconfiguration, and enables IPv6 processing on an interface.

IPv6 EUI-64 Addressing

The switch's implementation of IPv6 supports EUI-64 addressing. EUI-64 applies an IPv6 address that is based on the MAC address of the interface. The EUI-64 identifiers from the MAC address are used as the least significant 64 bits of a unicast address.

To enable IPv6 EUI-64, use the **ipv6 address** command on page 31.3 specifying the optional **eui64** parameter for an interface.

When configuring SLAAC you must ensure that you set the prefix length to 64 bits on the switch that is advertising the RAs used for address configuration via SLAAC.

Prefix information received in an RA (Router Advertisement) will not be applied to form an IPv6 address via SLAAC unless the prefix length is 64. Since the EUI is 64 bits long, the IPv6 prefix of the advertising device must also be 64 bits. This prefix length setting and behavior is in accordance with RFC 4864, section 5.5.3.

IPv6 Link-local Addresses

The switch's implementation of IPv6 supports IPv6 link-local addresses without global addresses for communications within the local subnetwork. Switches do not forward packets to link-local addresses. To enable IPv6 link-local addresses, use the **ipv6 enable** command on page 31.7. **ipv6 enable** automatically configures an IPv6 link-local address on the interface and enables IPv6 processing on the interface.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the **Link-local addresses** glossary entry, and the **ipv6 enable** command for more information. Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the **ipv6 enable** command then it will not be removed using a **no ipv6 address** command.

IPv6 RA Guard

This section describes the switch's support for IPv6 RA Guard, and how to configure IPv6 RA Guard on the switch.

RA Guard Introduction

Router Advertisements (RAs) and Router Redirects are used to manage IPv6 networks. RA messages advertise a router's presence and specify network parameters that are used by hosts as part of address auto-configuration and setting next-hop routes for particular destinations.

RAs are periodically transmitted by switches allowing networks to be reconfigured by changes to the routers only. Switches can also send redirects to hosts suggesting that they use a different next-hop route for a particular traffic stream. But because the entire network configuration can be modified by what is contained in RAs and redirects, the network is vulnerable to rogue messages that are generated either through misconfiguration or due to a malicious attack.

RA Guard on the switch simply considers each of its ports as either trusted or untrusted. Any host connected to a port is considered trusted or untrusted depending on the port status. A trusted port will accept RAs and redirects and will forward RAs and redirects on trusted ports. An untrusted port will block and discard all RAs and redirects received from the untrusted host.

Enabling IPv6 RA Guard

The switch's implementation of IPv6 RA Guard is disabled by default. To enable IPv6 RA Guard on a port to block RAs from an untrusted host, use the **ipv6 nd raguard** command on page 31.16. Disable IPv6 RA Guard to allow RAs on a port using the **no ipv6 nd raguard** command.

Chapter 31: IPv6 Commands

Command List 31.2
clear ipv6 neighbors 31.2
ipv6 address 31.3
ipv6 address autoconfig 31.5
ipv6 enable 31.7
ipv6 forwarding 31.8
ipv6 nd managed-config-flag 31.9
ipv6 nd minimum-ra-interval 31.10
ipv6 nd other-config-flag 31.11
ipv6 nd prefix 31.12
ipv6 nd ra-interval 31.14
ipv6 nd ra-lifetime 31.15
ipv6 nd raguard 31.16
ipv6 nd reachable-time 31.17
ipv6 nd retransmission-time 31.18
ipv6 nd suppress-ra 31.19
ipv6 neighbor 31.20
ipv6 opportunistic-nd 31.21
ipv6 route 31.22
ping ipv6 ..... 31.23
show ipv6 forwarding 31.24
show ipv6 interface brief 31.25
show ipv6 neighbors 31.25
show ipv6 route 31.26
show ipv6 route summary 31.28
traceroute ipv6 ..... 31.29

Command List

This chapter provides an alphabetical reference of commands used to configure IPv6. For more information, see Chapter 30, IPv6 Introduction.

clear ipv6 neighbors

Use this command to clear all dynamic IPv6 neighbor entries.

Syntax `clear ipv6 neighbors`

Mode Privileged Exec

Example

```
awplus# clear ipv6 neighbors
```

ipv6 address

Use this command to set the IPv6 address of a VLAN interface and enable IPv6.

Use the optional eui64 parameter to derive the interface identifier of the IPv6 address from the MAC address of the interface. Note that the MAC address of the default VLAN is applied if the interface does not have a MAC address of its own when specifying the eui64 parameter.

Use the no variant of this command to remove the IPv6 address assigned and disable IPv6. Note that if no global addresses are left after removing the IPv6 address then IPv6 is disabled.

Syntax

```
ipv6 address <ipv6-addr/prefix-length> [eui64]
no ipv6 address <ipv6-addr/prefix-length> [eui64]
```

Parameter	Description
<ipv6-addr/prefix-length>	Specifies the IPv6 address to be set. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64. Note that your switch will not accept prefix lengths greater than 64.
`[eui64]`	A method of automatically deriving the lower 64 bits of an IPv6 address, based on the switch's MAC address.

Mode Interface Configuration for a VLAN interface.

Usage If the **eui64** parameter is specified then the lower 64 bits of the IPv6 address are appended with the same address that would be acquired through stateless address autoconfiguration (SLAAC) if the device received an RA (Router Advertisement) specifying this prefix. See **ipv6 address autoconfig** for a detailed command description and examples to enable and disable SLAAC. See also the **IPv6 EUI-64 Addressing** section in **Chapter 30, IPv6 Introduction** for further EUI-64 implementation information.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the **Link-local addresses** glossary entry, and the **ipv6 enable** command for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the **ipv6 enable** command then it will not be removed using a **no ipv6 address** command.

Examples

To assign the IPv6 address `2001:0db8::a2/64` to the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 address 2001:0db8::a2/64
```

To remove the IPv6 address `2001:0db8::a2/64` from the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 address 2001:0db8::a2/64
```

To assign the **eui64** derived address in the prefix `2001:db8::/48` to VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-fr-subif)# ipv6 address 2001:0db8::/48 eui64
```

To remove the **eui64** derived address in the prefix `2001:db8::/48` from VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-fr-subif)# no ipv6 address 2001:0db8::/48 eui64
```

Validation Commands

show running-config
show ipv6 interface brief
show ipv6 route

Related Commands

ipv6 address autoconfig

ipv6 address autoconfig

Use this command to enable IPv6 stateless address autoconfiguration (SLAAC) for an interface. This configures an IPv6 address on an interface derived from the MAC address on the interface.

Use the **no** variant of this command to disable IPv6 SLAAC on an interface. Note that if no global addresses are left after removing all IPv6 autoconfigured addresses then IPv6 is disabled.

Syntax

```
ipv6 address autoconfig

no ipv6 address autoconfig
```

Mode Interface Configuration for a VLAN interface.

Usage The **ipv6 address autoconfig** command enables automatic configuration of IPv6 addresses using stateless autoconfiguration on an interface and enables IPv6, but does not enable IPv6 forwarding. See **ipv6 forwarding** command on page 31.8 for further description and examples.

IPv6 hosts can configure themselves when connected to an IPv6 network using ICMPv6 (Internet Control Message Protocol version 6) router discovery messages. Configured routers respond with a Router Advertisement (RA) containing configuration parameters for IPv6 hosts.

The SLAAC process derives the interface identifier of the IPv6 address from the MAC address of the interface. When applying SLAAC to an interface, note that the MAC address of the default VLAN is applied to the interface if the interface does not have its own MAC address.

If SLAAC is not suitable then a network can use stateful configuration with DHCPv6 (Dynamic Host Configuration Protocol version 6) Relay, or hosts can be configured statically. See **ip dhcp-relay server-address** for the DHCPv6 Relay server command description and examples. For introduction and configuration information about DHCPv6 Relay agent see **"DHCP Relay Agent Introduction" on page 89.9** and **"Configuring the DHCP Relay Agent" on page 89.9**.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the **Link-local addresses** glossary entry, and the **ipv6 enable** command for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the **ipv6 enable** command then it will not be removed using a **no ipv6 address** command.

Examples To enable SLAAC on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 address autoconfig
```

To disable SLAAC on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 address autoconfig
```

Validation Commands
show running-config
show ipv6 interface brief
show ipv6 route

Related Commands
ipv6 address
ipv6 enable

ipv6 enable

Use this command to enable IPv6 on an interface without an IPv6 global address for the interface. This enables IPv6 with a IPv6 link-local address, not an IPv6 global address.

Use the no variant of this command to disable IPv6 on an interface without a global address. Note the no variant of this command does not operate on an interface with an IPv6 global address or an interface configured for IPv6 stateless address autoconfiguration (SLAAC),

Syntax

```
ipv6 enable
no ipv6 enable
```

Mode Interface Configuration for a VLAN interface.

Usage The **ipv6 enable** command automatically configures an IPv6 link-local address on the interface and enables the interface for IPv6 processing. Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the **Link-local addresses** glossary entry for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the **ipv6 enable** command then it will not be removed using a **no ipv6 address** command.

Examples To enable IPv6 with only a link-local IPv6 address on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
```

To disable IPv6 with only a link-local IPv6 address on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 enable
```

Validation Commands
show running-config
show ipv6 interface brief
show ipv6 route

Related Commands
ipv6 address
ipv6 address autoconfig

ipv6 forwarding

Use this command to turn on IPv6 unicast routing for IPv6 packet forwarding. Execute this command globally on your switch prior to issuing **ipv6 enable** on individual interfaces.

Use this **no** variant of this command to turn off IPv6 unicast routing for IPv6 packet forwarding. Note IPv6 unicast routing for IPv6 packet forwarding is disabled by default.

Note

Use this command to enable IPv6 unicast routing before configuring either RIPng or OSPFv3 IPv6 routing protocols and static or multicast IPv6 routing. IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Syntax

```
ipv6 forwarding
no ipv6 forwarding
```

Mode Global Configuration

Default IPv6 unicast forwarding is disabled by default.

Usage Enable IPv6 unicast forwarding globally for all interface on your switch with this command. Use the **no** variant of this command to disable IPv6 unicast forwarding globally for all interfaces on your switch.

IPv6 unicast forwarding allows switches to communicate with devices that are more than one hop away, providing that there is a route to the destination address. If IPv6 forwarding is not enabled then pings to addresses on devices that are more than one hop away will fail, even if there is a route to the destination address.

Examples To enable IPv6 unicast routing, use this command as shown below:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
```

To disable IPv6 unicast routing, use the **no** variant of this command as shown below:

```
awplus# configure terminal
awplus(config)# no ipv6 forwarding
```

Related Commands

ipv6 enable
ipv6 multicast-routing

ipv6 nd managed-config-flag

Use this command to set the managed address configuration flag, contained within the router advertisement field.

Setting this flag indicates the operation of a stateful autoconfiguration protocol such as DHCPv6 for address autoconfiguration, and that address information (i.e. the network prefix) and other (non-address) information can be requested from the switch.

An unset flag enables hosts receiving the advertisements to use a stateless autoconfiguration mechanism to establish their IPv6 addresses. The default is flag unset.

Use the **no** variant of this command to reset this command to its default of, *flag unset*.

Syntax

```
ipv6 nd managed-config-flag
no ipv6 nd managed-config-flag
```

Default Unset

Mode Interface Configuration for a VLAN interface.

Usage Advertisement flags will not be transmitted unless you have applied the **no ipv6 nd suppress-ra** command on page 31.19. This step is included in the example below.

Example To set the managed address configuration flag on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 nd managed-config-flag
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands

ipv6 nd suppress-ra
ipv6 nd prefix
ipv6 nd other-config-flag

ipv6 nd minimum-ra-interval

Use this command in Interface Configuration mode to set a minimum Router Advertisement (RA) interval for a VLAN interface.

Use the **no** variant of this command in Interface Configuration mode to remove the minimum RA interval for a VLAN interface.

Syntax

```
ipv6 nd minimum-ra-interval <seconds>
no ipv6 nd minimum-ra-interval [<seconds>]
```

Parameter	Description
<seconds>	Specifies the number of seconds between IPv6 Router Advertisements (RAs). Valid values are from 3 to 1350 seconds.

Default The RA interval for a VLAN interface is unset by default.

Mode Interface Configuration for a VLAN interface.

Examples To set the minimum RA interval for the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 nd minimum-ra-interval 60
```

To remove the minimum RA interval for the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 nd minimum-ra-interval 60
```

Related Commands

ipv6 nd ra-interval
ipv6 nd suppress-ra
ipv6 nd prefix
ipv6 nd other-config-flag

ipv6 nd other-config-flag

Use this command to set the **other** stateful configuration flag (contained within the router advertisement field) to be used for IPv6 address auto-configuration. This flag is used to request the router to provide information in addition to providing addresses.

Note

Setting the **ipv6 nd managed-config-flag** command on page 31.9 implies that the **ipv6 nd other-config-flag** will also be set.

Use **no** variant of this command to reset the value to the default.

Syntax

```
ipv6 nd other-config-flag
no ipv6 nd other-config-flag
```

Default Unset

Mode Interface Configuration for a VLAN interface.

Usage Advertisement flags will not be transmitted unless you have applied the **no ipv6 nd suppress-ra** command on page 31.19. This step is included in the example below.

Example To set the IPv6 other-config-flag on the VLAN interface `vlan4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd other-config-flag
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands

ipv6 nd suppress-ra
ipv6 nd prefix
ipv6 nd managed-config-flag

ipv6 nd prefix

Use this command in Interface Configuration mode for a VLAN interface to specify the IPv6 prefix information that is advertised by the router advertisement for IPv6 address auto-configuration.

Use the **no** parameter with this command to reset the IPv6 prefix for a VLAN interface in Interface Configuration mode.

Syntax

```
ipv6 nd prefix <ipv6-prefix/length>

ipv6 nd prefix <ipv6-prefix/length> [<valid-lifetime>]

ipv6 nd prefix <ipv6-prefix/length> <valid-lifetime>
   <preferred-lifetime> [no-autoconfig]

ipv6 nd prefix <ipv6-prefix/length> <valid-lifetime>
   <preferred-lifetime> off-link [no-autoconfig]

no ipv6 nd prefix [<ipv6-addr/prefix-length>|all]
```

Parameter	Description
<ipv6-prefix/length>	The prefix to be advertised by the router advertisement message. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64.The default is X:X::/64.
<valid-lifetime>	The the period during which the specified IPv6 address prefix is valid. This can be set to a value between 0 and 4294967295 seconds. The default is 2592000 (30 days). Note that this period should be set to a value greater than that set for the prefix preferred-lifetime.
<preferred-lifetime>	Specifies theIPv6 prefix preferred lifetime. This is the period during which the IPv6 address prefix is considered a current (undeprecated) value. After this period, the command is still valid but should not be used in new communications. Set to a value between 0 and 4294967295 seconds. The default is 604800 seconds (7 days). Note that this period should be set to a value less than that set for the prefix valid-lifetime.
`off-link`	Specify the IPv6 prefix off-link flag. The default is *flag set*.
`no-autoconfig`	Specify the IPv6 prefix no autoconfiguration flag. Setting this flag indicates that the prefix is not to be used for autoconfiguration. The default is *flag set*.
`all`	Specify all IPv6 prefixes associated with the VLAN interface.

Default Valid-lifetime default is 2592000 seconds (30 days). Preferred-lifetime default is 604800 seconds (7 days).

Usage This command specifies the IPv6 prefix flags that are advertised by the router advertisement message.

Examples The following example configures the switch to issue router advertisements on the VLAN interface `vlan4`, and advertises the address prefix of `2001:0db8::/64`.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd prefix 2001:0db8::/64
```

The following example configures the switch to issue router advertisements on the VLAN interface `vlan4`, and advertises the address prefix of `2001:0db8::/64` with a valid lifetime of 10 days and a preferred lifetime of 5 days.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd prefix 2001:0db8::/64 864000 432000
```

The following example configures the switch to issue router advertisements on the VLAN interface `vlan4`, and advertises the address prefix of `2001:0db8::/64` with a valid lifetime of 10 days, a preferred lifetime of 5 days and no prefix used for autoconfiguration.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd prefix 2001:0db8::/64 864000 43200
no-autoconfig
```

The following example resets router advertisements on the VLAN interface `vlan4`, so the address prefix of `2001:0db8::/64` is not advertised from the switch.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd prefix 2001:0db8::/64
```

The following example resets all router advertisements on the VLAN interface `vlan4`:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd prefix all
```

Related Commands **ipv6 nd suppress-ra**

ipv6 nd ra-interval

Use this command to specify the interval between IPv6 Router Advertisements (RA) transmissions.

Use **no** parameter with this command to reset the value to the default value (600 seconds).

Syntax

```
ipv6 nd ra-interval <seconds>

no ipv6 nd ra-interval
```

Parameter	Description
<seconds>	Specifies the number of seconds between IPv6 Router Advertisements (RAs). Valid values are from 4 to 1800 seconds.

Default 600 seconds.

Mode Interface Configuration for a VLAN interface.

Usage Advertisement flags will not be transmitted unless you have applied the **no ipv6 nd suppress-ra** command on page 31.19 as shown in the example below.

Example To set the advertisements interval on the VLAN interface `vlan4` to be 60 seconds, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd ra-interval 60
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands

ipv6 nd minimum-ra-interval
ipv6 nd suppress-ra
ipv6 nd prefix

ipv6 nd ra-lifetime

Use this command to specify the time period that this router can usefully act as a default gateway for the network. Each router advertisement resets this time period.

Use **no** parameter with this command to reset the value to default.

Syntax

```
ipv6 nd ra-lifetime <seconds>
no ipv6 nd ra-lifetime
```

Parameter	Description
<seconds>	Time period in seconds. Valid values are from 0 to 9000. Note that you should set this time period to a value greater than the value you have set using the **ipv6 nd ra-interval** command.

Default 1800 seconds

Mode Interface Configuration for a VLAN interface.

Usage This command specifies the lifetime of the current router to be announced in IPv6 Router Advertisements.

Advertisement flags will not be transmitted unless you have applied the **no ipv6 nd suppress-ra** command. This instruction is included in the example shown below.

Examples To set the advertisement lifetime of 8000 seconds on the VLAN interface `vlan4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd ra-lifetime 8000
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands

ipv6 nd suppress-ra
ipv6 nd prefix

ipv6 nd raguard

Use this command to apply the Router Advertisements (RA) Guard feature from the Interface Configuration mode for a switch port. This blocks all RA messages received on a switch port. For introductory information about RA Guard see **"RA Guard Introduction" on page 30.13**.

Use the **no** parameter with this command to disable RA Guard for a specified switch port.

Syntax

```
ipv6 nd raguard
no ipv6 nd raguard
```

Default RA Guard is not enabled by default.

Mode Interface Configuration for a switch port interface.

Usage Router Advertisements (RAs) are used by Routers to announce themselves on the link. Applying RA Guard to a switch port disallows Router Advertisements and redirect messages. RA Guard blocks RAs from untrusted hosts. Blocking RAs stops untrusted hosts from flooding malicious RAs and stops any misconfigured hosts from disrupting traffic on the local network.

Enabling RA Guard on a port blocks RAs from a connected host and indicates the port and host are untrusted. Disabling RA Guard on a port allows RAs from a connected host and indicates the port and host are trusted. Ports and hosts are trusted by default to allow RAs.

Example To enable RA Guard on switch ports `port1.0.2-1.0.12`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-1.0.12
awplus(config-if)# ipv6 nd raguard
```

To verify RA Guard is enabled on switch port interface `port1.0.2`, use the command:

```
awplus# show running-config interface port1.0.2
```

To disable RA Guard on switch ports `port1.0.2-1.0.12`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-port1.0.12
awplus(config-if)# no ipv6 nd raguard
```

When RA Guard is disabled on a switch port it is not displayed in **show running-config** output.

Output **Example output from a show running-config interface port1.0.2 to verify RA Guard:**

```
!
interface port1.0.2
 switchport mode access
 ipv6 nd raguard
!
```

Related Commands **show running-config interface**

ipv6 nd reachable-time

Use this command to specify the reachable time in the router advertisement to be used for detecting reachability of the IPv6 neighbor.

Use the **no** variant of this command to reset the value to default.

Syntax

```
ipv6 nd reachable-time <milliseconds>
no ipv6 nd reachable-time
```

Parameter	Description
<milliseconds>	Time period in milliseconds. Valid values are from 1000 to 3600000. Setting this value to 0 indicates an unspecified reachable-time.

Default 0 milliseconds

Mode Interface Configuration for a VLAN interface.

Usage This command specifies the reachable time of the current router to be announced in IPv6 Router Advertisements.

Advertisement flags will not be transmitted unless you have applied the **no ipv6 nd suppress-ra** command. This instruction is included in the example shown below.

Example To set the reachable-time in router advertisements on the VLAN interface `vlan4` to be 1800000 milliseconds, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 nd reachable-time 1800000
awplus(config-if)# no ipv6 nd suppress-ra
```

To reset the reachable-time in router advertisements on the VLAN interface `vlan4` to an unspecified reachable-time (0 milliseconds), enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd reachable-time
```

Related Commands **ipv6 nd suppress-ra**
ipv6 nd prefix

ipv6 nd retransmission-time

Use this command to specify the advertised retransmission interval for Neighbor Solicitation in milliseconds between IPv6 Routers.

Use the **no** variant of this command to reset the retransmission time to the default (1 second).

Syntax

```
ipv6 nd retransmission-time <milliseconds>
no ipv6 nd retransmission-time [<milliseconds>]
```

Parameter	Description
<milliseconds>	Time period in milliseconds. Valid values are from 1000 to 3600000.

Default 1000 milliseconds (1 second)

Mode Interface Configuration for a VLAN interface.

Examples To set the retransmission-time of Neighbor Solicitation on the VLAN interface `vlan2` to be 800000 milliseconds, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 nd retransmission-time 800000
```

To reset the retransmission-time of Neighbor Solicitation on the VLAN interface `vlan2` to the default 1000 milliseconds (1 second), enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 nd retransmission-time
```

Related Commands **ipv6 nd suppress-ra**
ipv6 nd prefix

ipv6 nd suppress-ra

Use this command to inhibit IPv6 Router Advertisement (RA) transmission for the current interface. Router advertisements are used when applying IPv6 stateless auto-configuration.

Use **no** parameter with this command to enable Router Advertisement transmission.

Syntax

```
ipv6 nd suppress-ra
no ipv6 nd suppress-ra
```

Default Router Advertisement (RA) transmission is suppressed by default.

Mode Interface Configuration for a VLAN interface.

Example To enable the transmission of router advertisements from the VLAN interface `vlan4` on the switch, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd suppress-ra
```

Related Commands **ipv6 nd ra-interval**
ipv6 nd prefix

ipv6 neighbor

Use this command to add a static IPv6 neighbor entry.

Use the **no** variant of this command to remove a specific IPv6 neighbor entry.

Syntax

```
ipv6 neighbor <ipv6-address> <vlan-name> <mac-address> <port-list>
no ipv6 neighbor <ipv6-address> <vlan-name> <port-list>
```

Parameter	Description
<ipv6-address>	Specify the neighbor's IPv6 address in format `X:X::X:X`.
<vlan-name>	Specify the neighbor's VLAN name.
<mac-address>	Specify the MAC hardware address in hexadecimal notation with the format `HHHH.HHHH.HHHH`.
<port-list>	Specify the port number, or port range.

Mode Global Configuration

Usage Use this command to clear a specific IPv6 neighbor entry. To clear all dynamic address entries, use the **clear ipv6 neighbors** command.

Example To create a static neighbor entry for IPv6 address `2001:0db8::a2`, on `vlan 4`, MAC address `0000.cd28.0880`, on `port1.0.6`, use the command:

```
awplus# configure terminal
awplus(config)# ipv6 neighbor 2001:0db8::a2 vlan4
0000.cd28.0880 port1.0.6
```

Related Commands **clear ipv6 neighbors**

ipv6 opportunistic-nd

Use this command to enable opportunistic neighbor discovery for the global IPv6 ND cache. Opportunistic neighbor discovery changes the behavior for unsolicited ICMPv6 ND packet forwarding on the switch.

Use the **no** variant of this command to disable opportunistic neighbor discovery for the global IPv6 ND cache.

Syntax

```
ipv6 opportunistic-nd
no ipv6 opportunistic-nd
```

Default Opportunistic neighbor discovery is disabled by default.

Mode Global Configuration

Usage When opportunistic neighbor discovery is enabled, the switch will reply to any received unsolicited ICMPv6 ND packets. The source MAC address for the unsolicited ICMPv6 ND packet is added to the IPv6 ND cache, so the switch forwards the ICMPv6 ND packet. When opportunistic neighbor discovery is disabled, the source MAC address for the ICMPv6 packet is not added to the IPv6 ND cache, so the ICMPv6 ND packet is not forwarded by the switch.

Examples To enable opportunistic neighbor discovery for the IPv6 ND cache, enter:

```
awplus# configure terminal
awplus(config)# ipv6 opportunistic-nd
```

To disable opportunistic neighbor discovery for the IPv6 ND cache, enter:

```
awplus# configure terminal
awplus(config)# no ipv6 opportunistic-nd
```

Related Commands **arp opportunistic-nd**
show ipv6 neighbors

Validation Commands **show running-config interface**

ipv6 route

Use this command to establish the distance for static routes of a network prefix.

Use the **no** variant of this command to disable the distance for static routes of the network prefix.

Syntax

```
ipv6 route <dest-prefix> <dest-prefix/length> {<gateway-ip>|
   <gateway-name>} [<distvalue>]

no ipv6 route <dest-prefix> <dest-prefix/length> {<gateway-ip>|
   <gateway-name>} [<distvalue>]
```

Parameter	Description
<dest-prefix/length>	Specifies the IP destination prefix. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64.
<gateway-ip>	Specifies the IP gateway (or next hop) address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
<distvalue>	Specifies the administrative distance for the route. Valid values are from 1 to 255.
<gateway-name>	Specifies the name of the gateway (or next hop) interface.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ipv6 route 2001:0db8::1/128 myintname 32
```

Validation Commands **show running-config**
show ipv6 route

ping ipv6

This command sends a query to another IPv6 host (send Echo Request messages).

Note Use of the `interface` parameter keyword, plus an interface or an interface range, with this command is only valid when pinging an IPv6 link local address.

Syntax

```
ping ipv6 {<host>|<ipv6-address>} [repeat {<1-2147483647>|
    continuous}] [size <10-1452>] [interface <interface-list>]
    [timeout <1-65535>]
```

Parameter	Description
<ipv6-addr>	The destination IPv6 address. The IPv6 address uses the format X:X::X:X.
<hostname>	The destination hostname.
`repeat`	Specify the number of ping packets to send.
<1-2147483647>	Specify repeat count. The default is 5.
`size` *<10-1452>*	The number of data bytes to send, excluding the 8 byte ICMP header. The default is 56 (64 ICMP data bytes).
`interface` *<interface-list>*	The interface or range of configured IP interfaces to use as the source in the IP header of the ping packet.
`timeout` *<1-65535>*	The time in seconds to wait for echo replies if the ARP entry is present, before reporting that no reply was received. If no ARP entry is present, it does not wait.
`repeat`	Specify the number of ping packets to send.
<1-2147483647>	Specify repeat count. The default is 5.
`continuous`	Continuous ping.
`size` *<10-1452>*	The number of data bytes to send, excluding the 8 byte ICMP header. The default is 56 (64 ICMP data bytes).
`timeout` *<1-65535>*	The time in seconds to wait for echo replies if the ARP entry is present, before reporting that no reply was received. If no ARP entry is present, it does not wait.

Mode User Exec and Privileged Exec

Example

```
awplus# ping ipv6 2001:0db8::a2
```

Related Commands traceroute ipv6

show ipv6 forwarding

Use this command to display IPv6 forwarding status.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 forwarding
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show ipv6 forwarding
```

Output

Figure 31-1: Example output from the show ipv6 forwarding command

```
ipv6 forwarding is on
```

show ipv6 interface brief

Use this command to display brief information about interfaces and the IPv6 address assigned to them.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ipv6 interface [brief]`

Parameter	Description
`brief`	Specify this optional parameter to display brief IPv6 interface information.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ipv6 interface brief
```

Output **Figure 31-2: Example output from the show ipv6 interface brief command**

```
awplus#show ipv6 interface brief
Interface     IPv6-Address                                  Status      Protocol
lo            unassigned                                    admin up    running
vlan1         2001:db8::1/48                                admin up    down
              fe80::215:77ff:fee9:5c50/64
```

Related Commands show interface brief

show ipv6 neighbors

Use this command to display all IPv6 neighbors.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ipv6 neighbors`

Mode User Exec and Privileged Exec

show ipv6 route

Use this command to display the IPv6 routing table for a protocol or from a particular table.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 route [connected|database|ospf|rip|static|summary|
    <ipv6-address>|<ipv6-addr/prefix-length>)]
```

Parameter	Description
`connected`	Displays only the routes learned from connected interfaces.
`database`	Displays only the IPv6 routing information extracted from the database.
`ospf`	Displays only the routes learned from IPv6 Open Shortest Path First (OSPFv3).
`rip`	Displays only the routes learned from IPv6 Routing Information Protocol (RIPng).
`static`	Displays only the IPv6 static routes you have configured.
`summary`	Displays summary information from the IPv6 routing table.
`<ipv6-address>`	Displays the routes for the specified address in the IP routing table. The IPv6 address uses the format X:X::X:X/ Prefix-Length. The prefix-length is usually set between 0 and 64.
`<ipv6-prefix/length>`	Displays only the routes for the specified IP prefix.

Mode User Exec and Privileged Exec

Example 1 To display an IP route with all parameters turned on, use the following command:

```
awplus# show ipv6 route
```

Output **Figure 31-3: Example output of the show ipv6 route command**

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF
S    ::/0 [1/0] via 2001::a:0:0:c0a8:a6, vlan10
C    2001:db8::a:0:0:0:0/64 via ::, vlan10
C    2001:db8::14:0:0:0:0/64 via ::, vlan20
C    2001:db8::0:0:0:0/64 via ::, vlan30
C    2001:db8::28:0:0:0:0/64 via ::, vlan40
C    2001:db8::fa:0:0:0:0/64 via ::, vlan250
C    2001:db8::/64 via ::, vlan250
C    2001:db8::/64 via ::, vlan40
C    2001:db8::/64 via ::, vlan20
C    2001:db8::/64 via ::, vlan10
```

Example 2 To display all database entries for an IP route, use the following command:

```
awplus# show ipv6 route database
```

Output **Figure 31-4: Example output of the show ipv6 route database command**

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF
       > - selected route, * - FIB route, p - stale info
Timers: Uptime

S    ::/0 [1/0] via 2001::a:0:0:c0a8:a01 inactive, 6d22h12m
          [1/0] via 2001::fa:0:0:c0a8:fa01 inactive, 6d22h12m
```

show ipv6 route summary

Use this command to display the summary of the current NSM RIB entries.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 route summary
```

Mode

User Exec and Privileged Exec

Example

To display IP route summary, use the following command:

```
awplus# show ipv6 route summary
```

Output

Figure 31-5: Example output from the show ipv6 route summary command

```
IPv6 routing table name is Default-IPv6-Routing-Table(0)
IPv6 routing table maximum-paths is 4
RouteSource     Networks
connected       4
ospf            5
Total           9
FIB             5
```

Related Commands

show ip route
show ip route database

traceroute ipv6

Use this command to trace the route to the specified IPv6 host.

Syntax `traceroute ipv6 {<ipv6-addr>|<hostname>}`

Parameter	Description
<ipv6-addr>	The destination IPv6 address. The IPv6 address uses the format X:X::X:X.
<hostname>	The destination hostname.

Mode User Exec and Privileged Exec

Example To run a traceroute for the IPv6 address 2001:0db8::a2, use the following command:

```
awplus# traceroute ipv6 2001:0db8::a2
```

Related Commands ping ipv6

Chapter 32: IPv6to4 Tunneling Configuration

Introduction 32.2
6to4 Automatic Tunnel Configuration 32.2
Tunneling Operation 32.2
6to4 Tunnels, Operation and Configuration Summary 32.3
Example 6to4 Configuration 32.4

Introduction

This chapter contains a basic IPv6to4 automatic tunneling configuration example for reference.

To see details on the Tunneling commands used in this sample configuration, or to see the outputs of the Validation commands, refer to **Chapter 33, IPv6to4 Tunneling Commands**. Links to the Tunneling commands used in the examples are also provided below the examples.

6to4 Automatic Tunnel Configuration

IPv6 transition is required to migrate from IPv4 to IPv6. One method to connect to the global IPv6 network over the existing IPv4 network is called 6to4 automatic tunneling.

Although this method is called '6to4 tunneling', it does not involve discrete point-to-point tunnels. The 'tunneling' in '6to4 tunneling' refers to the fact that the IPv6 packets are encapsulated in IPv4 packets to be 'tunneled' across the IPv4 domain. Hence, '6to4 tunneling' is primarily a scheme for encapsulating IPv6 packets inside IPv4 headers.

Using 6to4 tunneling, you are not required to specify tunnel destination addresses. You only choose the interface that connects the switch to the IPv4 domain, and designate that interface to be the tunnel entry-point. As will be explained below, the IPv4 address that represents the point at which any given IPv6 packet will eventually exit the IPv4 domain is derived from the IPv6 packet itself. Note that the packet's destination IPv6 address itself contains the destination IPv4 address that is used for tunnel encapsulation.

The 6to4 tunneling implementation in AlliedWare Plus™ is used for routing only between IPv6 addresses that are of the form:

`2002: <valid-IPv4-unicast-address> : XX:XX:XX:XX:XX`

6to4 tunneling makes use of the fact that for every valid IPv4 unicast address *A.B.C.D*, there is always a corresponding valid IPv6 subnet `2002:<A.B.C.D>::/48`. So, for every global IP v4 address that has been allocated to an organization, there is immediately a global IPv6 subnet `2002:<ipv4-address>::/48` available to that organization.

Tunneling Operation

When an IPv4 interface on the switch is designated as a tunnel entry point, using the **tunnel source** command on page 33.5 command, an IPv6 tunnel interface is created. The interface is automatically allocated the IPv6 address: `2002:<ipv4-address>::1/128`.

Note Other implementations of 6to4 tunneling might have a different scheme for deriving the IPv6 address of the tunnel interface, as there is no standardized scheme. This does not cause any interoperability problems, however, as the IPv6 address of the tunnel interface is not actually involved in the routing process.

This tunnel interface is the gateway used by the IPv6 packets to enter the IPv4 domain. As IPv6 packets enter the IPv4 domain, the switch must encapsulate them by pre-pending IPv6 packets with an IPv4 header. The source address of the encapsulating header is the IPv4 address that has been specified by the tunnel source command. The process for determining the destination IP address is explained in more detail below.

The key to routing across the IPv4 domain is that there is a connection between the IPv4 address of a tunnel endpoint and the IPv6 subnets which can be reached via that endpoint. The connection is that the tunnel interface with IP address `<ipv4-address>` is considered to be the gateway to **all** IPv6 subnets within the range `2002:<ipv4-address>::/48`.

When a switch is required to deliver packets, via a tunnel interface, to IPv6 addresses in the range `2002:<ipv4-address>::/48`, the switch knows that the IPv4 address to which the switch must deliver that packet is given by the 17th through 48th bits of the IPv6 destination address. So, the encapsulation process extracts those bits from the IPv6 destination address, and uses them as the IPv4 destination address in the encapsulating header.

For example, if `192.0.2.1` is the IPv4 address of a tunnel endpoint, then the IPv6 subnets in the range `2002:c000:201::/48` are automatically known to be reachable via that tunnel endpoint. If another tunnel endpoint has an IPv6 packet to deliver to `2002:c000:201:6::04:8d`, via the tunnel, then that switch will encapsulate the packet in an IPv4 header with a destination address of `192.0.2.1`

The corollary to this is the requirement that, to be reachable from other 6to4 networks, the IPv6 VLAN interfaces on the IPv6 side a tunneling switch will need to be configured with IPv6 addresses that are subnetted from the `2002:<ipv4-address>::/48` address.

For example, `2002:<ipv4-address>:1::/64, 2002:<ipv4-address>:2::/64, 2002:<ipv4-address>:3::/64`

Note `2002:<ipv4-address>::/64` cannot be used here, because it conflicts with the automatically configured tunnel address.

6to4 Tunnels, Operation and Configuration Summary

1. When an IPv4 address is designated as the source address of a tunnel, that tunnel interface is automatically given the IPv6 address `2002:<ipv4-address>::1/128`

2. The switch does not automatically create any IPv6 routes via that tunnel interface, so you do need to explicitly create a route to direct traffic over the tunnel interface. Typically, you will create a route to `2002::/16` via the tunnel interface. There might be occasions, though, when you want to just create some more restricted routes, within the `2002::/16` range, over the tunnel interface.

3. Once routes have been created that direct traffic over the tunnel interface, it is not necessary to specify next hop addresses for the individual remote IPv6 subnets that are reachable via the tunnel. The switch will automatically know the IPv4 next hop via which to reach any IPv6 subnet in the range `2002:<ipv4-address>::/48`.

4. The only IPv6 addresses that can be routed to via the 6to4 tunnels on switches running AlliedWare Plus™ are IPv6 addresses of the form:
   `2002:<valid-IPv4-unicast-address>:xx:xx:xx:xx:xx`

Example 6to4 Configuration

The following example shows the minimum configuration required for 6to4 automatic tunnel configuration. Follow the commands and descriptions in the tables below the sample network:

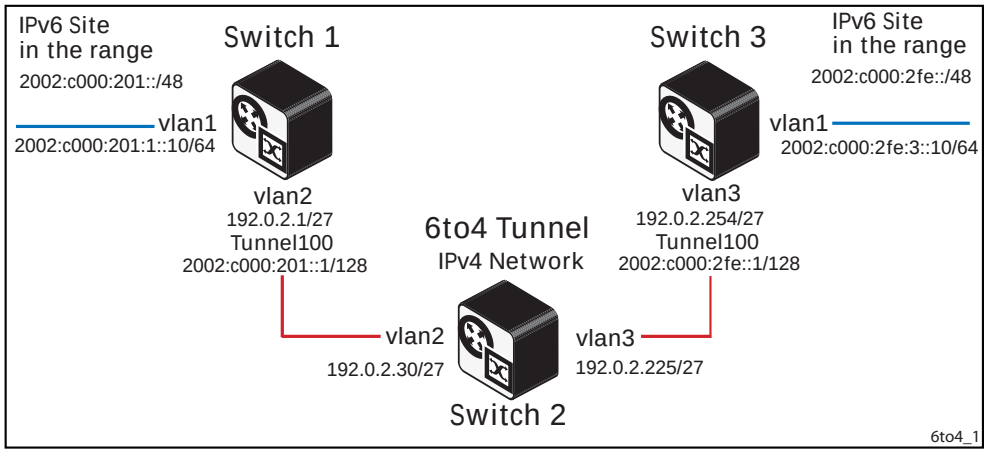


Table 32-1: Example configuration for 6to4 automatic tunneling: Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) and enter the Interface Configuration mode.
`awplusa(config-if)#` `ip address 192.0.2.1/27`	Set the IPv4 address for the interface (`vlan2`).
`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `interface vlan1`	Specify the interface (`vlan1`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ipv6 address 2002:c000:201:1::10/64`	Set the IPv6 address for the interface (`vlan1`).
`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `ipv6 forwarding`	Enable IPv6 routing.

Table 32-1: Example configuration for 6to4 automatic tunneling: Switch 1 (cont.)

`awplus(config)#` `interface tunnel 100`	Create a tunnel interface.
`awplus(config-if)#` `tunnel mode ipv6ip 6to4`	Set the tunnel mode.
`awplus(config-if)#` `tunnel source 192.0.2.1`	Define the IPv4 address to be used as the source address for the tunnel interface.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `ipv6 route 2002::/16 tunnel100`	Add a route to send traffic for other 6to4 subnets via the tunnel interface.

Table 32-2: Example configuration for 6to4 automatic tunneling: Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ip address 192.0.2.30/27`	Set the IPv4 address of the interface (`vlan2`).
`awplus(config-if)#` `exit`	Exit Interface mode and enter the Global Configuration mode.
`awplus(config)#` `interface vlan3`	Specify the interface (`vlan3`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ip address 192.0.2.225/27`	Set the IPv4 address of the interface (`vlan3`).
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Table 32-3: Example configuration for 6to4 automatic tunneling: Switch 3

`awplus(config)#` `interface vlan3`	Specify interface (`vlan3`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ip address 192.0.2.254/27`	Set the IPv4 address for the interface (`vlan3`).
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.
`awplus(config)#` `interface vlan1`	Specify the interface (`vlan1`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ipv6 address 2002:c000:2fe:3::10/64`	Set the IPv6 address for the interface (`vlan1`).
`awplus(config-if)#` `exit`	Exit Interface mode and enter Global Configuration mode.
`awplus(config)#` `ipv6 forwarding`	Enable IPv6 routing.
`awplus(config)#` `interface tunnel 100`	Create a tunnel interface.
`awplus(config-if)#` `tunnel mode ipv6ip 6to4`	Set the tunnel mode.
`awplus(config-if)#` `tunnel source 192.0.2.254`	Define the IPv4 address to be used as the source address for the tunnel interface.

Table 32-3: Example configuration for 6to4 automatic tunneling: Switch 3 (cont.)

`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `ipv6 route 2002::/16 tunnel100`	Add a route to send traffic for other 6to4 subnets via the tunnel interface.
`awplus(config)#` `router ospf`	Create an OSPF routing instance.
`awplus(config-router)#` `router-id 10.70.0.76`	Specify a Router ID for the OSPF routing process.
`awplus(config-router)#` `network 192.0.2.224/27 area 0`	Enable OSPF routing with a specified Area ID on interfaces with IP addresses within the specified network address. The Area ID used in this case is 0, which specifies the backbone area.

Note

The IPv6 address which is automatically assigned to the tunnel interface in this example for **Switch 1** is **2002:<IPv4 address in hex>::1/128**, where <**IPv4 address in hex**> represents the IPv4 address of the tunnel source, converted to hexadecimal.

So, the tunnel interface is the gateway from the IPv4 network to the IPv6 subnet
2002:<IPv4 address in hex>::/48

Hence, the IPv6 address configured on **vlan1** must be in the subnet **2002:<IPv4 address in hex>::/48**. However, if the address on **vlan1** is given a / **64** prefix-length it must differ from the tunnel address somewhere in the 49-64th bits.

For example, in this scenario: **vlan2** IP address (which the tunnel will go over): **192.0.2.1 = c0000201 hex (shown as c000:201without leading zeros in the example).** Tunnel 100 IPv6 address will automatically use **2002:c000:0201:0000::1/128** as its IPv6 address.

So, configure the following address on **vlan1: 2002:c000:0201:<xxxx>::x/6** where <**xxxx**> is anything other than **0000** e.g. **2002:c000:0201:1::10/64**

Names of commands used

interface tunnel
ip address
ipv6 address
tunnel mode ipv6ip
tunnel source

Chapter 33: IPv6to4 Tunneling Commands

Command List 33.2
interface tunnel 33.2
tunnel dscp (6to4) 33.3
tunnel mode ipv6ip 33.4
tunnel source 33.5
tunnel ttl 33.6

Command List

This chapter provides an alphabetical reference of commands used to configure automatic IPv6 tunneling over IPv4. For more information, see **Chapter 32, IPv6to4 Tunneling Configuration**.

interface tunnel

Use this command to create a new tunnel interface, which is identified by an integer (1-145).

This command is also used to enter interface configuration mode for existing tunnel interfaces.

Use the **no** variant of this command to destroy a previously created tunnel interface.

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCStack.

Syntax

```
interface tunnel <1-145>
no interface tunnel <1-145>
```

Default Disabled

Mode Global Configuration

Usage This command creates a new tunnel interface to configure in Interface mode.

Examples

```
awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)#

awplus# configure terminal
awplus(config)# no interface tunnel 100
awplus(config)#
```

Related Commands **tunnel dscp (6to4)**
tunnel source

tunnel dscp (6to4)

Use this command to configure the value (0-63) to use for the IPv4 DSCP (Differentiated Services Code Point) field in the IPv4 header that encapsulates the tunneled IPv6 packets. By default the IPv4 DSCP field value is 0.

Use the **no** variant of this command to reset the IPv4 DSCP field to the default (0).

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCStack.

Syntax

```
tunnel dscp <0-63>
no tunnel dscp
```

Default The default IPv4 DSCP field value is 0.

Mode Interface Configuration for 6to4 tunnel interfaces.

Usage This command controls the IPv4 DSCP field in the IPv4 headers that are prepended (or prefixed) to the tunneled IPv6 packets.

Examples

```
awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1
awplus(config-if)# tunnel dscp 10

awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1
awplus(config-if)# no tunnel dscp
```

Related Commands

interface tunnel
tunnel mode ipv6ip
tunnel source

tunnel mode ipv6ip

Use this command to specify the IPv6 transition tunnel mode. In AlliedWare Plus™ 6to4 automatic tunneling is the only tunnel mode supported.

Use the **no** variant of this command to return the mode of the IPv6 transition tunnel to an undefined state.

Note This command will function on your switch in the stand-alone mode, but is not supported when the switch forms part of a VCStack.

Syntax

```
tunnel mode ipv6ip 6to4

no tunnel mode
```

Parameter	Description
`6to4`	`6to4` automatic tunnel mode.

Mode Interface Configuration for 6to4 tunnel interfaces.

Usage This command specifies a tunnel encapsulation mode for IPv6 in IPv4. Currently only 6to4 automatic tunneling is supported. Future releases may support alternative tunneling modes.

It is a requirement that the mode is set on a tunnel. The tunnel will not be operational until the tunnel mode has been set using the **tunnel mode ipv6ip 6to4** command.

Examples

```
awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1
awplus(config-if)# tunnel mode ipv6ip

awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1
awplus(config-if)# no tunnel mode
```

Related Commands

interface tunnel
tunnel dscp (6to4)
tunnel source

tunnel source

Use this command to specify the IPv4 source address for packets being encapsulated in the tunnel. It should be the IPv4 address on the interface that attaches the switch for the IPv4 domain through which the IPv6 packets are to be tunneled.

Use the **no** variant of this command to remove a tunnel source address for a tunnel interface.

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCStack.

Syntax

```
tunnel source <ipv4-addr>
no tunnel source
```

Parameter	Description
<ipv4-addr>	IPv4 tunnel source address.

Mode Interface Configuration for 6to4 tunnel interfaces.

Usage This command specifies an IPv4 source address for the tunnel.

Note There are constraints on the IPv4 source address specified for the tunnel. The IPv4 source address for the tunnel must be the IPv4 address of an interface on the switch, and it must be the interface for carrying the tunnel traffic.

Examples

```
awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1

awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# no tunnel source
```

Related Commands interface tunnel
tunnel dscp (6to4)

tunnel ttl

Use this command to configure the value to use for the Time to Live (TTL) field in the IPv4 header that encapsulates the tunneled IPv6 packets.

Use the **no** variant of this command to inherit the TTL value from the encapsulated packet.

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCStack.

Syntax

```
tunnel ttl <1-255>
no tunnel ttl
```

Default By default the TTL value is inherited from the encapsulated packet.

Mode Interface Configuration for 6to4 tunnel interfaces.

Usage This command specifies a value of Time to Live (TTL) in the tunnel IPv4 encapsulation header.

Examples

```
awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1
awplus(config-if)# tunnel ttl 255

awplus# configure terminal
awplus(config)# interface tunnel 100
awplus(config-if)# tunnel source 192.0.2.1
awplus(config-if)# no tunnel ttl
```

Related Commands **interface tunnel**
tunnel dscp (6to4)

Chapter 34: Routing Protocol Overview

Introduction 34.2
RIP 34.2
OSPF 34.2
VRRP 34.3

Introduction

This chapter introduces the basic routing protocols supported within the AlliedWare Plus™ Operating System.

RIP

A distance-vector protocol, Routing Information Protocol (RIP) is an Interior Gateway Protocol (IGP) that uses hop counts as its metrics. The AlliedWare Plus™ RIP module supports RFCs 1058 and 1723; the RIPv2 module supports more fields in the RIP packets, and supports security authentication features.

At regular intervals of the routing update timer (which has a default value of 30 seconds), and at the time of change in the topology, the RIP router sends update messages to other routers. The listening routers update their route table with the new route, and increase the metric value of the path by one (referred to as a hop count). The router recognizes the IP address advertising router as the next hop, then sends the routing updates to other routers. A maximum allowable hop count is 15. If a router reaches a metric value of 16 or more (referred to as infinity), the destination is identified as unreachable. This avoids the indefinite routing loops. The split-horizon and hold-down features are used to avoid propagation of incorrect routing information. The route becomes not valid when the route time-out timer expires; it remains in the table until the route-flush timer expires.

OSPF

A link-state routing protocol, Open Shortest Path First (OSPF) is an Interior Gateway Protocol (IGP) that uses the Shortest Path First (SPF) Dijsktra algorithm.

OSPF sends link-state advertisements (LSAs) to all other routers within the same hierarchical area. Data on attached interfaces, metrics used, and other variables, are included in OSPF LSAs. As OSPF routers accumulate link-state data, they use the SPF algorithm to calculate the shortest path to each node.

An Autonomous System (AS) or Domain is defined as a group of networks with common routing infrastructure. OSPF can work in one AS; or receive or send routes from or to different AS systems. Autonomous systems consist of areas. An area is a group of neighboring networks or attached hosts. A router attached to multiple areas with its interfaces is called an Area Border Router (ABR). It creates a distinct topological database: a group of LSAs received from all routers in the same area, for each area. All the routers in the same area have an identical topological database. OSPF routing traffic is restricted in the area because areas are unknown to each other. The routing information is distributed between areas, area border routers, networks, and connected routers by the OSPF backbone.

All backbone OSPF area routers use the same procedures and algorithms to maintain routing information within the backbone that any area router would. The backbone topology is invisible to all routers within an area. The individual area topologies are invisible to the backbone. Sometimes the backbone is not a contiguous area. Virtual links function as if they were direct links, and are configured between backbone routers that share a link to a non-backbone area.

AS border routers running OSPF learn about exterior routes through exterior gateway protocols (EGPs) such as the Border Gateway Protocol (BGP).

During boot-up, an OSPF router initializes its routing-protocol-specific data structures and tables. When the lower layer protocols with which it interfaces are functional, it sends the OSPF Hello protocol packets to find neighboring routers. A router sends Hello packets as keep-alive packets, informing other routers about its continuing functionality. Two routers are adjacent when their link state databases are synchronized.

Multi-access networks have more than two routers. On multi-access networks, the hello protocol chooses a designated router and a designated backup-router. The designated router generates LSAs for the entire multi-access network, and reduces network traffic and the size of the topological database. The designated router also determines the adjacency of routers and the synchronization of their topological databases. The data on a router's adjacencies or state changes are provided by periodic transmission of an LSA. Failed routers are detected, and topology is changed quickly by comparison of adjacencies to link states. Each router calculates a shortest path tree, with itself as a root, from the topological database generated from these LSAs. This shortest path tree creates a routing table.

VRRP

Mission-critical applications running on fault-tolerant networking equipment, such as routers and switches, require redundancy and high availability. This section provides an architectural overview of Virtual Router Redundancy Protocol (VRRP) implementation in the AlliedWare Plus™ OS.

Typically, end hosts are connected to the enterprise network through a single router (first-hop router) that is in the same Local Area Network (LAN) segment. The most popular method of configuration is for the end hosts to statically configure this router as their default gateway. This minimizes configuration and processing overhead. The main problem with this configuration method is that it produces a single point of failure if the enterprise network's first-hop router fails.

VRRP attempts to solve this problem by introducing the concept of a virtual router, composed of two or more VRRP routers on the same subnet. The concept of a virtual IP address is also introduced, which is the address that end hosts configure as their default gateway. Only one of the routers (called the Master) forwards packets on behalf of this IP address. In the event that the Master fails, one of the other routers (Backups) assumes forwarding responsibility for it.

Chapter 35: Route Selection

Introduction 35.2
Types of Routes 35.2
- Interface Routes 35.2
- Static Routes 35.2
- Dynamic Routes 35.3

RIB and FIB Routing Tables 35.4
- Understanding the Routing Information Base (RIB) 35.4
- Administrative Distance 35.6
- Metric 35.8
- Equal Cost Multipath Routing 35.8
- How AlliedWare Plus Deletes Routes 35.8
- How AlliedWare Plus Adds Routes 35.9
- Troubleshooting Routes not Installed in the RIB 35.10
- Troubleshooting Routes not Installed in the FIB 35.10

Introduction

This chapter describes the route selection process used by the AlliedWare Plus™ Operating System. Understanding the route selection process helps in analyzing and troubleshooting route-related problems.

The process of routing packets consists of selectively forwarding data packets from one network to another. Your device must determine which network to send each packet to, and over which interface to send the packet in order to reach the desired network. This information is contained in your device routes. For each packet, your device chooses the best route it has for that packet and uses that route to forward the packet. In addition, you can define filters to restrict the way packets are sent.

Types of Routes

Your device learns routes from static information entered as part of the configuration process and by listening to any configured routing protocols. The following types of routes are available on your device:

Interface Routes

Your device creates an interface route when you create the interface. This route tells your device to send packets over that interface when the packets are addressed to the interface's subnet.

Static Routes

You can manually enter routes, which are then called static routes. You can use static routes to:

- specify the default route (to 0.0.0.0). If your device does not have a route to the packet's destination, it sends it out the default route. The default route normally points to an external network such as the Internet.
- set up multiple networks or subnets. In this case you define multiple routes for a particular interface, usually a LAN port. This is a method of supporting multiple subnets on a single physical media.

To create a static route, use the command:

```
awplus(config)# ip route <subnet&mask> {<gateway-ip>|
                <interface>} [<distance>]
```

Dynamic Routes

Your device learns dynamic routes from one or more routing protocols such as RIP, BGP, or OSPF. The routing protocol updates these routes as the network topology changes.

In all but the most simple networks, we recommend that you configure at least one dynamic routing protocol. Routing protocols enables your device to learn routes from other routers and switches on the network, and to respond automatically to changes in network topology.

Routing protocols use different metrics to calculate the best path for a destination. However, when two paths have an equal cost/metric and Equal Cost Multipath (ECMP) is enabled on a system, AlliedWare Plus™ may receive two paths from the same protocol.

- Routing Information Protocol (RIP)

  Routing Information Protocol (RIP) is a simple distance vector IPv4 routing protocol. It determines the number of hops between the destination and your device, where one hop is one link. Given a choice of routes, RIP uses the route that takes the lowest number of hops. If multiple routes have the same hop count, RIP chooses the first route it finds.

  See **Chapter 37, RIP Configuration** for further information about RIP Configuration.

- Routing Information Protocol next generation (RIPng)

  Routing Information Protocol next generation (RIPng) is a simple distance vector IPv6 routing protocol. It determines the number of hops between the destination and your device, where one hop is one link. Given a choice of routes, RIPng uses the route that takes the lowest number of hops. If multiple routes have the same hop count, RIPng chooses the first route it finds.

  RIPng (Routing Information Protocol next generation) is an extension of RIPv2 to support IPv6. RFC 2080 specifies RIPng. The differences between RIPv2 and RIPng are:

  « RIPng does not support RIP updates authentication

  « RIPng does not allow the attachment of arbitrary tags to routes

  « RIPng requires the encoding of the next-hop for a set of routes

  See **Chapter 39, RIPng for IPv6 Configuration** for further information about RIPng Configuration.

- Open Shortest Path First (OSPF)

  The Open Shortest Path First (OSPF) protocol is documented in RFC 1247. It has a number of significant benefits over RIP, including:

  « OSPF supports the concept of areas to allow networks to be administratively partitioned as they grow in size.

  « Load balancing, in which multiple routes exist to a destination, is also supported. OSPF distributes traffic over these links.

  See **Chapter 41, OSPF Introduction and Configuration** for further information about OSPF Configuration.

- Border Gateway Protocol (BGP)

  The Border Gateway Protocol (BGP) allows routers in different routing domains to exchange routing information. This facilitates the forwarding of data across the borders of the routing domains. BGP-4 is based on distance vector (DV) protocol algorithms.

RIB and FIB Routing Tables

Your device maintains its routing information in routing tables that tell your device how to find a remote network or host. Each route is uniquely identified in a table by its IP address, network mask, next hop, interface, protocol, and policy. There are two routing tables populated by your device: the **Routing Information Base** (RIB) and the **Forwarding Information Base** (FIB).

Note Routes in the FIB are used locally but are not advertised to neighbors if they are not also in the RIB.

Routing Information Base

The RIB records **all** the routes that your device has learned. Your device uses the RIB to advertise routes to its neighbor devices and to populate the FIB. It adds routes to this table when:

- you add a static route using the **ip route** command
- one or more routing protocols, such as RIP or OSPF, exchange routing information with other routers or hosts
- your device receives route information from a connected interface
- your device gathers route information from an ICMP redirect message or DHCP message

Understanding the Routing Information Base (RIB)

Use the **show ipv6 route database** command to view the IPv6 RIB.

Use the **show ip route database** command to view the IPv4 RIB.

The RIB in AlliedWare Plus displays all the routes sent to the RIB by the routing protocols, plus all the static and connected routes.

The angle bracket > character in show output indicates which route has been selected as the best route. The best routes are installed in the Forwarding Information Base (FIB).

Routes which have been installed in the software FIB are marked with a star * symbol in show output.

The Administrative Distance and the Metric are seen in the square brackets with **AD** on the left of the backslash and **Metric** to the right of **AD**, so this is shown as: **[AD / Metric]**

See the below list of other information displayed in the RIB:

- Route type
- Prefix and Prefix Length
- Administrative Distance
- Metric
- Next-Hop
- Exit interface
- Uptime

Example RIB Output

See the sample output below for example RIB output, and note that all routes (including the non-best routes) are displayed in the RIB, but note only the best routes are selected:

Figure 35-1: Example RIB output after entering the show ipv6 route database command

```
awplus#show ipv6 route database
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
> - selected route, * - FIB route, p - stale info
Timers: Uptime
R *> 2001:db8:10::/64 [120/2] via fe80::eecd:6dff:fe20:c26b, vlan20, 20:42:47
R 2001:db8:20::/64 [0/1] via ::, vlan20, 21:18:42
C *> 2001:db8:20::/64 via ::, vlan20, 21:18:42
O 2001:db8:40::/64 [110/1] via ::, vlan40, 21:18:21
C *> 2001:db8:40::/64 via ::, vlan40, 21:18:22
O *> 2001:db8:50::/64 [110/2] via fe80::eecd:6dff:fe20:c073, vlan40, 21:17:29
O *> 2001:db8:60::/64 [110/2] via fe80::eecd:6dff:fe20:c073, vlan40, 20:31:06
```

Forwarding Information Base

The RIB populates the **Forwarding Information Base** (FIB) with the best route to each destination. When your device receives an IP packet, and no filters are active that would exclude the packet, it uses the FIB to find the most specific route to the destination. If your device does not find a direct route to the destination, and no default route exists, it discards the packet and sends an ICMP message to that effect back to the source.

A route is only deleted in the FIB if the corresponding prefix is removed from the RIB.

Changes to the software FIB are propagated to the hardware FIB, so the software and hardware FIB tables mirror each other.

The number of FIB table entries also mirror the best selected routes in the RIB.

Example FIB Output

See the sample output below for example FIB output, and note that only the best selected routes from the RIB are installed in the FIB:

Figure 35-2: Example FIB output after entering the show ipv6 route command

```
awplus#show ipv6 route
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, D - DHCP
IA - OSPF inter area E1 - OSPF ext. type 1, E2 - OSPF ext. type 2
Timers: Uptime
R 2001:db8:10::/64 [120/2] via fe80::eecd:6dff:fe20:c26b, vlan20, 21:28:22
C 2001:db8:20::/64 via ::, vlan20, 22:04:17
C 2001:db8:40::/64 via ::, vlan40, 22:03:57
O 2001:db8:50::/64 [110/2] via fe80::eecd:6dff:fe20:c073, vlan40, 22:03:04
O 2001:db8:60::/64 [110/2] via fe80::eecd:6dff:fe20:c073, vlan40, 21:16:41
```

Viewing table entries

To view the routes in the RIB, use the commands:

```
awplus# show ip route database [connected|ospf|rip|static]
```

```
awplus# show ip route database
```

To view the routes in the FIB, use the command:

```
awplus# show ip route [connected|ospf|rip|static|<ip-addr>|
       <ip-addr/prefix-length>]
```

Administrative Distance

When multiple routes are available for the same prefix, the AlliedWare Plus™ Operating System adds the routes with the lowest **administrative distance** to the FIB. The administrative distance is a rank given to a route based on the protocol that the route was received from. The lower the administrative distance, the higher the route preference. For example, if the RIB has these routes:

Route	Prefix	Protocol	Distance
1	192.168.1.0/24	Static	1
2	192.168.2.0/24	eBGP	20
3	192.168.2.0/24	OSPF	110
4	192.168.3.0/24	OSPF	110

then the AlliedWare Plus™ Operating System adds routes 1, 2, and 4 to the FIB. It does not add route 3, as this has a higher administrative distance than a route with the same prefix.

Note Administrative distance indicates a level of trustworthiness of a route where the lower the administrative distance the higher the integrity of a route.

The following table lists the default administrative distances for routing protocols.

Protocols	Distance	Preference
Connected Routes directly connected to an interface.	-	1 (highest)
Static Routes added using the **ip route** command or learned through DHCP options on interfaces using DHCP to obtain an IP address.	1	2
eBGP Routes learned from BGP that are external to your network.	20	3
OSPF Routes learned from OSPF.	110	4
RIP Routes learned from RIP.	120	5
iBGP Routes learned from BGP that are internal to your network.	200	6 (lowest)
Unknown No traffic will be passed to neighbors via this route.	255	(route is not advertised to neighbors)

For static routes, specify the distance when adding the route, use the command:

```
awplus(config)# ip route <subnet&mask> [<gateway-ip>]
                [<interface>] [<distance>]
```

To enter a separate administrative distance value for each OSPF route type, enter the Router Configuration mode and use the command:

```
awplus(config-router)# distance ospf {external <1-255>|
                       inter-area <1-255>|intra-area <1-255>}
```

To set the same value for all OSPF route types, use the command:

```
awplus(config-router)# distance <1-255>
```

For RIP routes, enter the Router Configuration mode, and use the command:

```
awplus(config-router)# distance <1-255> [<ip-addr/prefix-
                       length> [<access-list>]]
```

This sets the administrative distance for all RIP routes.

You cannot set an administrative distance for connected routes.

Note

AlliedWare Plus™ does not populate routes with an administrative distance of 255 in the FIB (Forwarding Information Base). But AlliedWare Plus™ does populate routes with an administrative distance of 255 in the RIB (Routing Information Base).
See the below examples showing the behavior of a static route with an administrative distance of 255, which is only added to the RIB, as seen from the below show output.

Output **Figure 35-3: Static route with an administrative distance of 255 that is added to the RIB**

```
awplus(config)#ip route 100.0.0.0/24 192.168.1.100 255
awplus(config)#end
awplus#show ip route database

Codes: C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       > - selected route, * - FIB route, p - stale info

S       100.0.0.0/24 [255/0] via 192.168.1.100, vlan1
C    *> 192.168.1.0/24 is directly connected, vlan1
```

Figure 35-4: Static route with an administrative distance of 255 that is not added to the FIB

```
awplus(config)#ip route 100.0.0.0/24 192.168.1.100 255
awplus(config)#end
awplus#show ip route

Codes: C - connected, S - static, R - RIP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       * - candidate default

C       192.168.1.0/24 is directly connected, vlan1
```

Metric

The Metric is used to find the best route from one routing source. In the routing table it is used as the next tie breaker if the Administrative Distance (AD) is equal for the routes in it.

Metrics used by AlliedWare Plus routing protocols

The following metrics are used by AlliedWare IPv4 and IPv6 routing protocols:

IPv4 and IPv6 Routing Protocol	Metric
RIP for IPv4 / RIPng for IPv6	Hop-Count
OSPFv2 for IPv4 / OSPFv3 for IPv6	Cost
BGP for IPv4 / BGP4+ for IPv6	MULTI_EXIT-DISC / IGMP metric to Next-Hop

Equal Cost Multipath Routing

When multiple routes are available for the same prefix within the FIB, then your device uses Equal Cost Multipath Routing (ECMP) to determine how to forward packets.

ECMP allows the AlliedWare Plus™ Operating System to distribute traffic over multiple equal-cost routes to a destination. The software determines that two or more routes are equal cost if they have the same destination IP address and mask. When the software learns such multiple routes, it puts them in an ECMP route group. When it sends traffic to that destination, it distributes the traffic across all routes in the group.

The AlliedWare Plus™ Operating System distributes traffic over the routes one flow at a time, so all packets in a session take the same route. Each equal-cost route group can contain up to eight individual routes. ECMP is only used to select between routes already in the FIB.

By default, each equal-cost route group can contain four routes. You can change this setting by using the command:

```
awplus(config)# maximum-paths <1-8>
```

The maximum path setting determines how many routes with the same prefix value and the same administrative distance that the FIB can contain. Once an equal-cost route group has the maximum number of routes, then the RIB cannot add any further routes to the route group. The device only adds to the group if a route is deleted from the FIB.

To disable ECMP, set the maximum paths value to one.

How AlliedWare Plus Deletes Routes

When the AlliedWare Plus™ Operating System receives a route delete request from a routing protocol, it first deletes the specified route from its RIB. Then it checks if the specified route is in the FIB.

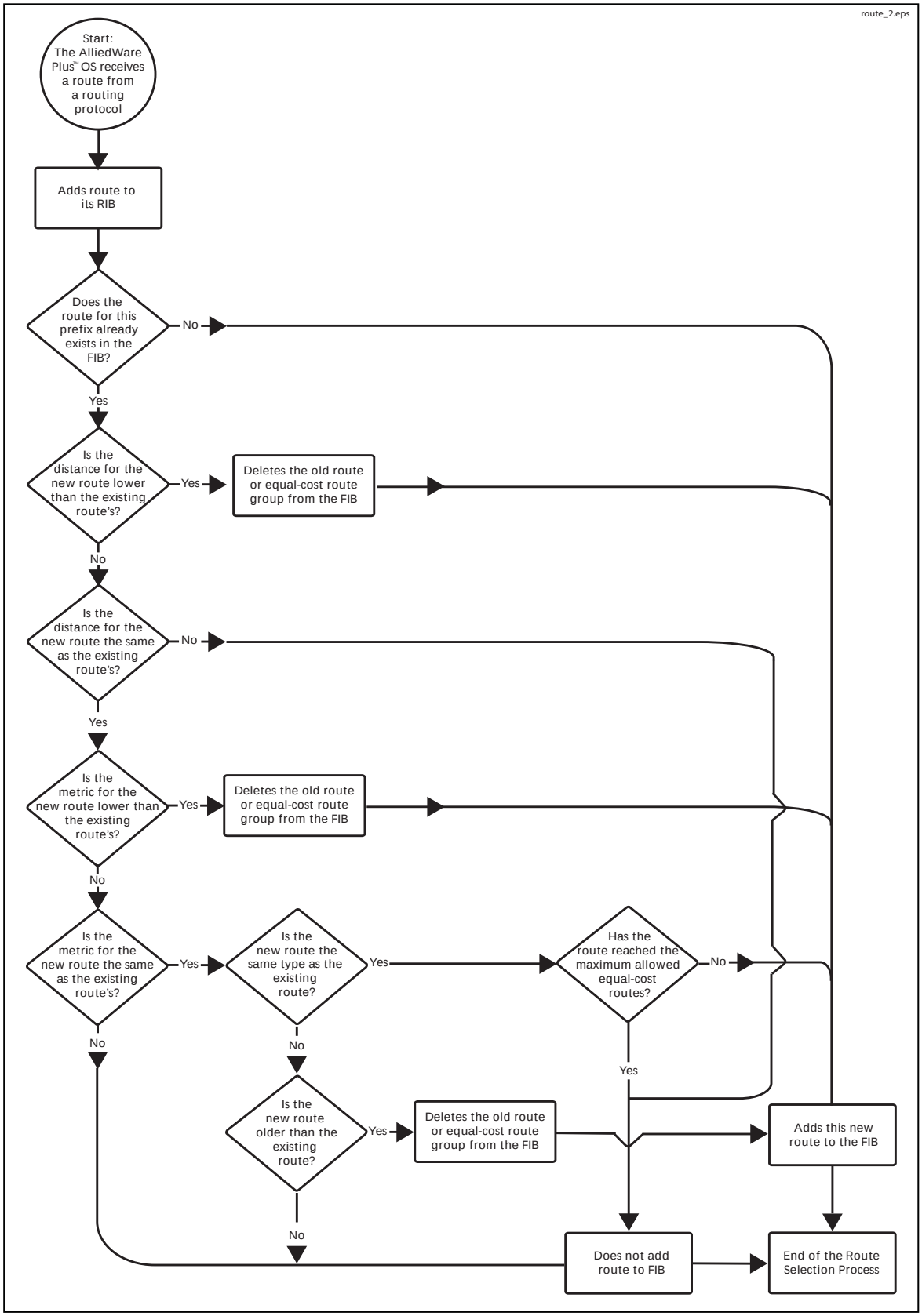
If the route is in the FIB, it deletes it from the FIB and checks if another route is available in its database for the same prefix.

If there is another route in the database, the software installs this route in the FIB. When multiple such routes exist, the software uses the route selection mechanism to choose the best route before adding it to the FIB.

How AlliedWare Plus Adds Routes

The following flow chart shows how the software adds a route to the FIB.

Figure 35-5: How AlliedWare Plus™ adds a route to the FIB



Troubleshooting Routes not Installed in the RIB

Possible reasons why a route is not installed in the RIB are:

- The layer 3 interface is not in the Up state.
- Route oscillation (route flap) is occurring with the route being added and removed frequently.
- The routing process from which the route is learned, has deleted the route.
- A routing protocol has learned the maximum number of routes allowed by the license, so the routes are not installed to the RIB.

See the **max-static-routes** command in the **System Configuration and Monitoring Commands** chapter for detailed command description and command example information, where static routes are applied before adding routes to the RIB.

Troubleshooting Routes not Installed in the FIB

Possible reasons why a route is not installed in the FIB are:

- The maximum-paths limit may have been reached (currently supports up to eight equal cost paths being installed).
- The maximum-paths command may be set to a lower value preventing more paths being selected as best.
- The desired route type has a higher AD over another route entry in the RIB, so is not preferred.
- The max-fib-routes command is configured and the maximum number of installed software FIB routes has been reached.

See the **max-fib-routes** command in the **System Configuration and Monitoring Commands** chapter for detailed command description and command example information to control the maximum number of FIB routes configured.

Chapter 36: Routing Commands

Introduction 36.2
Command List 36.3
ip route 36.3
maximum-paths 36.5
show ip route 36.6
show ip route database 36.8
show ip route summary 36.10

Introduction

This chapter provides an alphabetical reference of routing commands that are common across the routing IP protocols.

For more information see **Chapter 34, Routing Protocol Overview** and **Chapter 35, Route Selection**.

Command List

ip route

This command adds a static route to the Routing Information Base (RIB). If this route is the best route for the destination, then your device adds it to the Forwarding Information Base (FIB). Your device uses the FIB to advertise routes to neighbors and forward packets.

The **no** variant of this command removes the static route from the RIB and FIB.

Syntax

```
ip route <subnet&mask> {<gateway-ip>|<interface>} [<distance>]
no ip route <subnet&mask> {<gateway-ip>|<interface>} [<distance>]
```

Parameter	Description
<subnet&mask>	The IPv4 address of the destination subnet defined using either a prefix length or a separate mask specified in one of the following formats:
	The IPv4 subnet address in dotted decimal notation followed by the subnet mask, also in dotted decimal notation.
	The IPv4 subnet address in dotted decimal notation, followed by a forward slash, then the prefix length.
<gateway-ip>	The IPv4 address of the gateway device.
<interface>	The interface that connects your device to the network. Enter the name of the VLAN or its VID. You can also enter 'null' as an interface. Specify a 'null' interface to add a null or blackhole route to the switch. The gateway IP address or the interface is required.
<distance>	The administrative distance for the static route in the range <1-255>. Static routes by default have an administrative distance of 1.

Mode Global Configuration

Default The default administrative distance for a static route is 1 for priority over non-static routes.

Usage Administrative distance can be modified so static routes do not take priority over other routes.

Specify a 'Null' interface to add a null or blackhole route to the switch. A null or blackhole route is a routing table entry that does not forward packets, so any packets sent to it are dropped.

Examples

To add the destination 192.168.3.0 with the mask 255.255.255.0 as a static route available through the device at "10.10.0.2" with the default administrative distance, use the commands:

```
awplus# configure terminal
awplus(config)# ip route 192.168.3.0 255.255.255.0 10.10.0.2
```

To remove the destination 192.168.3.0 with the mask 255.255.255.0 as a static route available through the device at "10.10.0.2" with the default administrative distance, use the commands:

```
awplus# configure terminal
awplus(config)# no ip route 192.168.3.0 255.255.255.0 10.10.0.2
```

To specify a null or blackhole route 192.168.4.0/24, so packets forwarded to this route are dropped, use the commands:

```
awplus# configure terminal
awplus(config)# ip route 192.168.4.0/24 null
```

To add the destination 192.168.3.0 with the mask 255.255.255.0 as a static route available through the device at "10.10.0.2" with an administrative distance of 128, use the commands:

```
awplus# configure terminal
awplus(config)# ip route 192.168.3.0 255.255.255.0 10.10.0.2 128
```

Related Commands

show ip route
show ip route database

maximum-paths

This command enables ECMP on your device, and sets the maximum number of paths that each route has in the Forwarding Information Base (FIB). ECMP is enabled by default.

The **no** variant of this command sets the maximum paths to the default of 4.

Syntax

```
maximum-paths <1-8>
no maximum-paths
```

Parameter	Description
<1-8>	The maximum number of paths that a route can have in the FIB.

Default By default the maximum number of paths is 4.

Mode Global Configuration

Examples To set the maximum number of paths for each route in the FIB to 5, use the command:

```
awplus# configure terminal
awplus(config)# maximum-paths 5
```

To set the maximum paths for a route to the default of 4, use the command:

```
awplus# configure terminal
awplus(config)# no maximum-paths
```

show ip route

Use this command to display routing entries in the FIB (Forwarding Information Base). The FIB contains the best routes to a destination, and your device uses these routes when forwarding traffic. You can display a subset of the entries in the FIB based on protocol.

To modify the lines displayed, use the | (output modifier token); to save the output to a file, use the > output redirection token.

Syntax

```
show ip route [connected|ospf|rip|static|<ip-addr>|<ip-addr/prefix-
    length>]
```

Parameter	Description
`connected`	Displays only the routes learned from connected interfaces.
`ospf`	Displays only the routes learned from OSPF.
`rip`	Displays only the routes learned from RIP.
`static`	Displays only the static routes you have configured.
<ip-addr>	Displays the routes for the specified address. Enter an IPv4 address.
<ip-addr/prefix-length>	Displays the routes for the specified network. Enter an IPv4 address and prefix length.

Mode User Exec and Privileged Exec

Example To display the static routes in the FIB, use the command:

```
awplus# show ip route static
```

To display the OSPF routes in the FIB, use the command:

```
awplus# show ip route ospf
```

Output Each entry in the output from this command has a code preceding it, indicating the source of the routing entry. For example, O indicates OSPF as the origin of the route. The first few lines of the output list the possible codes that may be seen with the route entries.

Typically, route entries are composed of the following elements:

- code
- a second label indicating the sub-type of the route
- network or host ip address
- administrative distance and metric
- next hop ip address
- outgoing interface name
- time since route entry was added

Figure 36-1: Example output from the show ip route command

```
Codes: C - connected, S - static, R - RIP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       * - candidate default

O       10.10.37.0/24 [110/11] via 10.10.31.16, vlan2, 00:20:54
C       3.3.3.0/24 is directly connected, vlan1
C       10.10.31.0/24 is directly connected, vlan2
C       10.70.0.0/24 is directly connected, vlan4
O E2    14.5.1.0/24 [110/20] via 10.10.31.16, vlan2, 00:18:56
C       33.33.33.33/32 is directly connected, lo
```

To avoid repetition, only selected route entries comprising of different elements are described here:

OSPF Route

```
O       10.10.37.0/24 [110/11] via 10.10.31.16, vlan2, 00:20:54
```

This route entry denotes:

- This route in the network `10.10.37.0/24` was added by OSPF.
- This route has an administrative distance of 110 and metric/cost of 11.
- This route is reachable via next hop `10.10.31.16`.
- The outgoing local interface for this route is `vlan2`.
- This route was added 20 minutes and 54 seconds ago.

```
C       10.10.31.0/24 is directly connected, vlan2
```

This route entry denotes:

- Route entries for network `10.10.31.0/24` are derived from the IP address of local interface `vlan2`.
- These routes are marked as Connected routes `(C)` and always preferred over routes for the same network learned from other routing protocols.

OSPF External Route

```
O E2    14.5.1.0/24 [110/20] via 10.10.31.16, vlan2, 00:18:56
```

This route entry denotes:

- This route is the same as the other OSPF route explained above; the main difference is that it is a Type 2 External OSPF route.

Related Commands

maximum-paths
show ip route database

show ip route database

This command displays the routing entries in the RIB (Routing Information Base).

When multiple entries are available for the same prefix, RIB uses the routes' administrative distances to choose the best route. All best routes are entered into the FIB (Forwarding Information Base). To view the routes in the FIB, use the **show ip route** command.

To modify the lines displayed, use the | (output modifier token); to save the output to a file, use the > output redirection token.

Syntax `show ip route database [connected|ospf|rip|static]`

Parameter	Description
`connected`	Displays only the routes learned from connected interfaces.
`ospf`	Displays only the routes learned from OSPF.
`rip`	Displays only the routes learned from RIP.
`static`	Displays only the static routes you have configured.

Mode User Exec and Privileged Exec

Example To display the static routes in the RIB, use the command:

```
awplus# show ip route database static
```

Output **Figure 36-2: Example output from the show ip route database command**

```
Codes: C - connected, S - static, R - RIP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
              > - selected route, * - FIB route, p - stale info

O    *> 9.9.9.9/32 [110/31] via 10.10.31.16, vlan2, 00:19:21
O       10.10.31.0/24 [110/1] is directly connected, vlan2, 00:28:20
C    *> 10.10.31.0/24 is directly connected, vlan2
S    *> 10.10.34.0/24 [1/0] via 10.10.31.16, vlan2
O       10.10.34.0/24 [110/31] via 10.10.31.16, vlan2, 00:21:19
O    *> 10.10.37.0/24 [110/11] via 10.10.31.16, vlan2, 00:21:19
C    *> 10.30.0.0/24 is directly connected, vlan6
S    *> 11.22.11.0/24 [1/0] via 10.10.31.16, vlan2
O E2 *> 14.5.1.0/24 [110/20] via 10.10.31.16, vlan2, 00:19:21
O       16.16.16.16/32 [110/11] via 10.10.31.16, vlan2, 00:21:19
S    *> 16.16.16.16/32 [1/0] via 10.10.31.16, vlan2
O    *> 17.17.17.17/32 [110/31] via 10.10.31.16, vlan2, 00:21:19
C    *> 45.45.45.45/32 is directly connected, lo
O    *> 55.55.55.55/32 [110/21] via 10.10.31.16, vlan2, 00:21:19
C    *> 127.0.0.0/8 is directly connected, lo
```

The routes added to the FIB are marked with a *. When multiple routes are available for the same prefix, the best route is indicated with the > symbol. All unselected routes have neither the * nor the > symbol.

```
S    *> 10.10.34.0/24 [1/0] via 10.10.31.16, vlan2
O       10.10.34.0/24 [110/31] via 10.10.31.16, vlan2, 00:21:19
```

These route entries denote:

- The same prefix was learned from OSPF and from static route configuration.
- Since this static route has a lower administrative distance than the OSPF route (110), the static route (1) is selected and installed in the FIB.

If the static route becomes unavailable, then the device automatically selects the OSPF route and installs it in the FIB.

Related Commands **maximum-paths**
show ip route

show ip route summary

This command displays a summary of the current RIB (Routing Information Base) entries.

To modify the lines displayed, use the | (output modifier token); to save the output to a file, use the > output redirection token.

Syntax `show ip route summary`

Mode User Exec and Privileged Exec

Example To display a summary of the current RIB entries, use the command:

```
awplus# show ip route summary
```

Output **Figure 36-3: Example output from the show ip route summary command**

```
IP routing table name is Default-IP-Routing-Table(0)
IP routing table maximum-paths is 4
Route Source    Networks
connected       5
ospf            2
Total           8
```

Related Commands **show ip route**
show ip route database

Chapter 37: RIP Configuration

Introduction 37.2
Enabling RIP 37.2
Specifying the RIP Version 37.4
RIPv2 Authentication (Single Key) 37.6
RIPv2 Text Authentication (Multiple Keys) 37.8
RIPv2 md5 authentication (Multiple Keys) 37.12

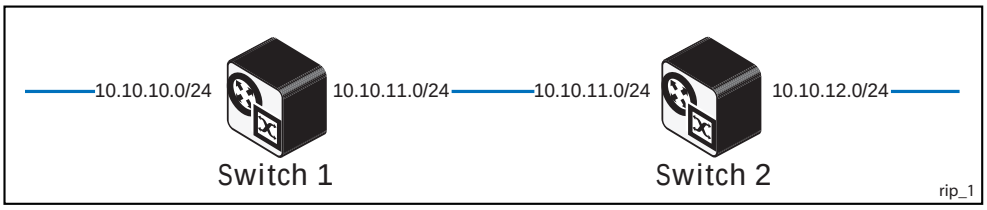
Introduction

This chapter contains basic RIP configuration examples. To see details on the RIP commands used in these examples, or to see the outputs of the validation commands, refer to the **Chapter 38, RIP Commands**.

Enabling RIP

This example shows the minimum configuration required for enabling two devices to exchange routing information using RIP. The routing devices in this example are Allied Telesis managed Layer 3 Switches. `Switch 1` and `Switch 2` are two neighbors connecting to network 10.10.11.0/24. Switch 1 and Switch 2 are also connected to networks 10.10.10.0/24 and 10.10.12.0/24 respectively. This example assumes that the devices have already been configured with IP interfaces in those subnets.

To enable RIP, first define the RIP routing process then associate a network with the routing process.



Switch 1

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#`	
`network 10.10.10.0/24`	Associate network 10.10.10.0/24 with the RIP process.
`awplus#`	
`network 10.10.11.0/24`	Associate network 10.10.11.0/24 with the RIP process.

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#` `network 10.10.11.0/24`	Associate networks with the RIP process
`awplus(config-router)#` `network 10.10.12.0/24`	Associate networks with the RIP process

Names of commands used

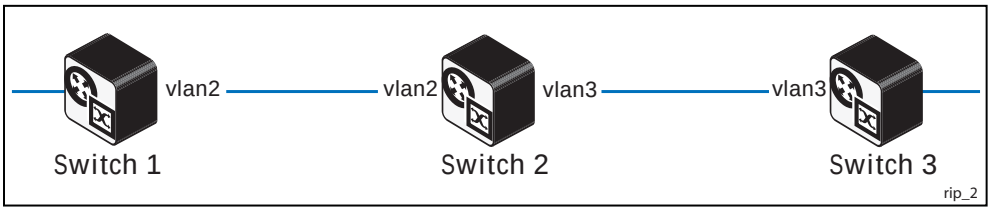
router rip
network (RIP)

Validation commands

show ip rip
show running-config
show ip protocols rip
show ip rip interface
show ip route

Specifying the RIP Version

Configure a router to receive and send specific versions of RIP packets on a VLAN interface. The routing devices in this example are Allied Telesis managed Layer 3 Switches. In this example, Switch 2 is configured to receive and send RIP version 1 and version 2 information on both `vlan2` and `vlan3` interfaces.



Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router rip`	Enable the RIP routing process.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode
`awplus(config)#` `interface vlan2`	Specify vlan2 as an interface you want to configure.
`awplus(config-if)#` `ip rip send version 1 2`	Allow sending RIP version 1 and version 2 packets out of this interface.
`awplus(config-if)#` `ip rip receive version 1 2`	Allow receiving of RIP version 1 and version 2 packets from the vlan2 interface.
`awplus(config-if)#` `exit`	Exit the Interface mode and return to Global Configuration mode to configure the next interface.
`awplus(config)#` `interface vlan3`	Specify interface vlan3 as the interface you want to configure.
`awplus(config-if)#` `ip rip send version 1 2`	Allow sending RIP version 1 and version 2 packets out of this interface.
`awplus(config-if)#` `ip rip receive version 1 2`	Allow receiving of RIP version 1 and version 2 packets from the vlan3 interface.

Names of commands used

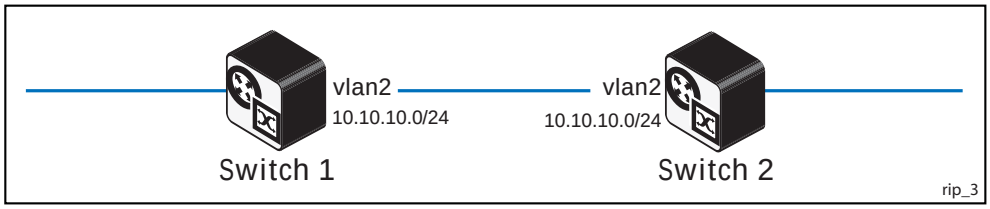
ip rip send version
ip rip receive version

Validation commands

show ip rip
show running-config
show ip protocols rip
show ip rip interface
show ip route

RIPv2 Authentication (Single Key)

AlliedWare Plus™ RIP implementation provides the choice of configuring authentication for a single key or for multiple keys. This example illustrates authentication of the routing information exchange process for RIP using a single key. The routing devices in this example are Allied Telesis managed Layer 3 Switches. Switch 1 and Switch 2 are running RIP and exchange routing updates. To configure single key authentication on Switch 1, specify an interface and then define a key or password for that interface. Next, specify an authentication mode. Any receiving RIP packet on this specified interface should have the same string as password. For an exchange of updates between Switch 1 and Switch 2, define the same password and authentication mode on Switch 2.



Switch 1

`awplus#` `configure terminal`	Enter the Configure mode.
`awplus(config)#` `router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#` `network 10.10.10.0/24`	Associate network 10.10.10.0/24 with the RIP process.
`awplus(config-router)#` `redistribute connected`	Enable redistributing from connected routes.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Configure mode.
`awplus(config)#` `interface vlan2`	Specify the VLAN interface (vlan2) for authentication.
`awplus(config-if)#` `ip rip authentication string Secret`	Specify the authentication string (Secret) for this interface.
`awplus(config-if)#` `ip rip authentication mode md5`	Specify the authentication mode to be MD5.

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#` `network 10.10.10.0/24`	Associate network 10.10.10.0/24 with the RIP process.
`awplus(config-router)#` `redistribute connected`	Enable redistributing from connected routes.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the VLAN interface (vlan2) for authentication.
`awplus(config-if)#` `ip rip authentication string Secret`	Specify the authentication string (Secret) on this interface.
`awplus(config-if)#` `ip rip authentication mode md5`	Specify the authentication mode to be MD5.

Names of commands used

ip rip authentication string
ip rip authentication mode
redistribute (RIP)
network (RIP)

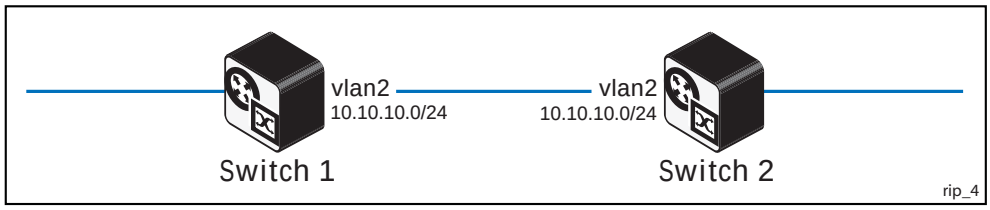
Validation commands

show ip rip
show running-config
show ip protocols rip
show ip rip interface
show ip route

RIPv2 Text Authentication (Multiple Keys)

This example illustrates text authentication of the routing information exchange process for RIP using multiple keys. The routing devices in this example are Allied Telesis managed Layer 3 Switches. Switch 1 and Switch 2 are running RIP and exchanging routing updates. To configure authentication on Switch 1, define a key chain, specify keys in the key chain and then define the authentication string or passwords to be used by the keys. Set the time period during which it is valid to receive or send the authentication key by specifying the accept and send lifetimes. After defining the key string, specify the key chain (or the set of keys) that will be used for authentication on each interface and also the authentication mode to be used.

Switch 1 accepts all packets that contain any key string that matches one of the key strings included in the specified key chain (within the accept lifetime) on that interface. The key ID is not considered for matching. For additional security, the accept lifetime and send lifetime are configured such that every fifth day the key ID and key string changes. To maintain continuity, the accept lifetimes should be configured to overlap. This will accommodate different time-setup on machines. However, the send lifetime does not need to overlap and we recommend not configuring overlapping send lifetimes.



Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#` `network 10.10.10.0/24`	Associate network 10.10.10.0/24 with the RIP process.
`awplus(config-router)#` `redistribute connected`	Enable redistributing of connected routes.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `key chain SUN`	Enter the key chain management mode to add keys to the key chain SUN.
`awplus(config-keychain)#` `key 10`	Add authentication key ID (10) to the key chain SUN.
`awplus(config-keychain-key)#` `key-string Secret`	Specify a password (Secret) to be used by the specified key.

Switch 1(cont.)

`awplus(config-keychain-key)#`	
`accept-lifetime 12:00:00 Mar 2 2007 14:00:00 Mar 7 2007`	Specify the time period during which authentication key string `Secret` can be received. In this case, key string `Secret` can be received from noon of March 2 to 2 pm March 7, 2007.
`awplus(config-keychain-key)#`	
`send-lifetime 12:00:00 Mar 2 2007 12:00:00 Mar 7 2007`	Specify the time period during which authentication key string `Secret` can be send. In this case, key string `Secret` can be received from noon of March 2 to noon of March 7, 2007.
`awplus(config-keychain-key)#`	
`exit`	Exit the keychain-key mode and return to keychain mode.
`awplus(config-keychain)#`	
`key 20`	Add another authentication key (`20`) to the key chain `SUN`.
`awplus(config-keychain-key)#`	
`key-string Earth`	Specify a password (`Earth`) to be used by the specified key.
`awplus(config-keychain-key)#`	
`accept-lifetime 12:00:00 Mar 7 2007 14:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be received. In this case, key string `Earth` can be received from noon of March 7 to 2 pm March 12, 2007.
`awplus(config-keychain-key)#`	
`send-lifetime 12:00:00 Mar 7 2007 12:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be sent. In this case, key string `Secret` can be received from noon of March 7 to noon of March 12, 2007.
`awplus(config-keychain-key)#`	
`end`	Enter Privileged Exec mode.
`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`interface vlan2`	Specify VLAN interface (vlan2) as the interface you want to configure on Switch 1.
`awplus(config-if)#`	
`ip rip authentication key-chain SUN`	Enable RIPv2 authentication on the `vlan2` interface and specify the key chain `SUN` to be used for authentication.
`awplus(config-if)#`	
`ip rip authentication mode text`	Specify text authentication mode to be used for RIP packets. This step is optional, as text is the default mode.

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#` `network 10.10.10.0/24`	Associate network 10.10.10.0/24 with the RIP process.
`awplus(config-router)#` `redistribute connected`	Enable redistributing from connected routes.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `key chain MOON`	Enter the key chain management mode to add keys to the key chain `MOON`.
`awplus(config-keychain)#` `key 30`	Add authentication key ID (`30`) to the key chain `MOON`.
`awplus(config-keychain-key)#` `key-string Secret`	Specify a password (`Secret`) to be used by the specified key.
`awplus(config-keychain-key)#` `accept-lifetime 12:00:00 Mar 2 2007 14:00:00 Mar 7 2007`	Specify the time period during which authentication key string `Secret` can be received. In this case, key string `Secret` can be received from noon of March 2 to 2 pm March 7, 2007.
`awplus(config-keychain-key)#` `send-lifetime 12:00:00 Mar 2 2007 12:00:00 Mar 7 2007`	Specify the time period during which authentication key string `Secret` can be send. In this case, key string `Secret` can be received from noon of March 2 to noon of March 7, 2007.
`awplus(config-keychain)#` `key 40`	Add another authentication key (`40`) to the key chain `MOON`.
`awplus(config-keychain-key)#` `key-string Earth`	Specify a password (`Earth`) to be used by the specified key.
`awplus(config-keychain-key)#` `accept-lifetime 12:00:00 Mar 7 2007 14:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be received. In this case, key string `Earth` can be received from noon of March 7 to 2 pm March 12, 2007.

Switch 2(cont.)

`awplus(config-keychain-key)#`	
`send-lifetime 12:00:00 Mar 7 2007 12:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be sent. In this case, key string `Secret` can be received from noon of March 7 to noon of March 12, 2007.
`awplus(config-keychain-key)#`	
`end`	Enter Privileged Exec mode.
`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`interface vlan2`	Specify the VLAN interface that you want to configure on Switch 2.
`awplus(config-if)#`	
`ip rip authentication key-chain MOON`	Enable RIPv2 authentication on the vlan2 interface, and specify the key chain MOON to be used for authentication.
`awplus(config-if)#`	
`ip rip authentication mode text`	Specify authentication mode to be used for RIP packets. This step is optional, as text is the default mode.

Names of commands used

key chain, **key-string**
accept-lifetime
send-lifetime
ip rip authentication key-chain
ip rip authentication mode

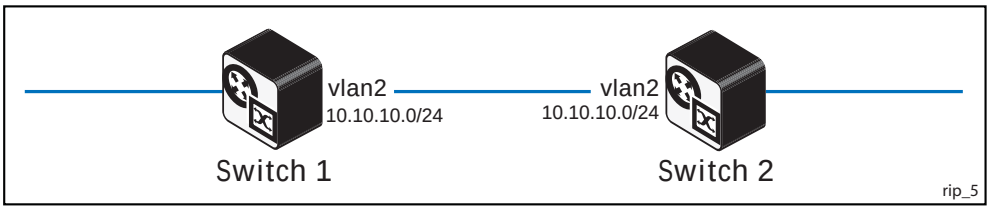
Validation commands

show ip rip
show running-config
show ip protocols rip
show ip rip interface
show ip route

RIPv2 md5 authentication (Multiple Keys)

This example illustrates the md5 authentication of the routing information exchange process for RIP using multiple keys. The routing devices in this example are Allied Telesis managed Layer 3 Switches. Switch 1 and Switch 2 are running RIP and exchange routing updates. To configure authentication on Switch 1, define a key chain, specify keys in the key chain and then define the authentication string or passwords to be used by the keys. Then set the time period during which it is valid to receive or send the authentication key by specifying the accept and send lifetimes. After defining the key string, specify the key chain (or the set of keys) that will be used for authentication on the interface and the authentication mode to be used. Configure Switch 2 and Switch 3 to have the same key ID and key string as Switch 1 for the time that updates need to be exchanged.

In `md5` authentication, both the key ID and key string are matched for authentication. Switch 1 will receive only packets that match both the key ID and the key string in the specified key chain (within the accept lifetime) on that interface. In the following example, Switch 2 has the same key ID and key string as Switch 1. For additional security, the accept lifetime and send lifetime are configured such that every fifth day the key ID and key string changes. To maintain continuity, the accept lifetimes should be configured to overlap; however, the send lifetime should not be overlapping.



Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router rip`	Define a RIP routing process and enter the Router Configuration mode.
`awplus(config-router)#` `network 10.10.10.0/24`	Associate network `10.10.10.0/24` with the RIP process.
`awplus(config-router)#` `redistribute connected`	Enable redistributing from connected routes.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `key chain SUN`	Enter the key chain management mode to add keys to the key chain `SUN`.
`awplus(config-keychain)#` `key 1`	Add authentication key ID (`1`) to the key chain `SUN`.

Switch 1(cont.)

`awplus(config-keychain-key)#` `key-string Secret`	Specify a password (`Secret`) to be used by the specified key.
`awplus(config-keychain-key)#` `accept-lifetime 12:00:00 Mar 2 2007 14:00:00 Mar 7 2007`	Specify the time period during which authentication key string `Secret` can be received. In this case, key string `Secret` can be received from noon of March 2 to 2 pm March 7, 2007.
`awplus(config-keychain-key)#` `send-lifetime 12:00:00 Mar 2 2007 12:00:00 Mar 7 2007`	Specify the time period during which authentication key string `Secret` can be send. In this case, key string `Secret` can be received from noon of March 2 to noon of March 7, 2007.
`awplus(config-keychain-key)#` `exit`	Exit the `keychain-key` mode and return to `keychain` mode.
`awplus(config-keychain)#` `key 2`	Add another authentication key (`2`) to the key chain `SUN`.
`awplus(config-keychain-key)#` `key-string Earth`	Specify a password (`Earth`) to be used by the specified key.
`awplus(config-keychain-key)#` `accept-lifetime 12:00:00 Mar 7 2007 14:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be received. In this case, key string `Earth` can be received from noon of March 7 to 2 pm March 12, 2007.
`awplus(config-keychain-key)#` `send-lifetime 12:00:00 Mar 7 2007 12:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be send. In this case, key string `Secret` can be received from noon of March 7 to noon of March 12, 2007.
`awplus(config-keychain-key)#` `end`	Enter Privileged Exec mode.
`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify interface `vlan2` as the VLAN interface you want to configure on Switch 1.
`awplus(config-if)#` `ip rip authentication key-chain SUN`	Enable RIPv2 authentication on the `vlan2` interface and specify the key chain `SUN` to be used for authentication.
`awplus(config-if)#` `ip rip authentication mode md5`	Specify the `md5` authentication mode to be used for RIP packets.

Switch 2

awplus# configure terminal	Enter the Global Configuration mode.
awplus(config)# router rip	Define a RIP routing process and enter the Router Configuration mode.
awplus(config-router)# network 10.10.10.0/24	Associate network 10.10.10.0/24 with the RIP process.
awplus(config-router)# redistribute connected	Enable redistributing from connected routes.
awplus(config-router)# exit	Exit the Router Configuration mode and return to the Global Configuration mode.
awplus(config)# key chain MOON	Enter the key chain management mode to add keys to the key chain MOON.
awplus(config-keychain)# key 1	Add authentication key ID (1) to the key chain MOON.
awplus(config-keychain-key)# key-string Secret	Specify a password (Secret) to be used by the specified key.
awplus(config-keychain-key)# accept-lifetime 12:00:00 Mar 2 2007 14:00:00 Mar 7 2007	Specify the time period during which authentication key string Secret can be received. In this case, key string Secret can be received from noon of March 2 to 2 pm March 7, 2007.
awplus(config-keychain-key)# send-lifetime 12:00:00 Mar 2 2007 12:00:00 Mar 7 2007	Specify the time period during which authentication key string Secret can be send. In this case, key string Secret can be received from noon of March 2 to noon of March 7, 2007.
awplus(config-keychain)# key 2	Add another authentication key (2) to the key chain MOON.
awplus(config-keychain-key)# key-string Earth	Specify a password (Earth) to be used by the specified key.
awplus(config-keychain-key)# accept-lifetime 12:00:00 Mar 7 2007 14:00:00 Mar 12 2007	Specify the time period during which authentication key string Earth can be received. In this case, key string Earth can be received from noon of March 7 to 2 pm March 12, 2007.

Switch 2(cont.)

`awplus(config-keychain-key)#`	
`send-lifetime 12:00:00 Mar 7 2007 12:00:00 Mar 12 2007`	Specify the time period during which authentication key string `Earth` can be send. In this case, key string `Secret` can be received from noon of March 7 to noon of March 12, 2007.
`awplus(config-keychain-key)#`	
`end`	Enter Privileged Exec mode.
`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`interface vlan2`	Specify vlan2 as the VLAN interface you want to configure on Switch 2.
`awplus(config-if)#`	
`ip rip authentication key-chain MOON`	Enable RIPv2 authentication on the `vlan2` interface and specify the key chain `MOON` to be used for authentication.
`awplus(config-if)#`	
`ip rip authentication mode md5`	Specify the md5 authentication mode to be used for RIP packets.

Names of commands used

key chain
key-string
accept-lifetime
send-lifetime
ip rip authentication key-chain
ip rip authentication mode

Validation commands

show ip rip
show running-config
show ip protocols rip
show ip rip interface

Chapter 38: RIP Commands

Introduction 38.2
Command List 38.2
accept-lifetime 38.2
alliedware-behavior 38.4
cisco-metric-behavior (RIP) 38.6
clear ip rip route 38.7
debug rip 38.8
default-information originate (RIP) 38.9
default-metric (RIP) 38.10
distance (RIP) 38.11
distribute-list (RIP) 38.12
fullupdate (RIP) 38.13
ip rip authentication key-chain 38.14
ip rip authentication mode 38.17
ip rip authentication string 38.21
ip rip receive-packet 38.23
ip rip receive version 38.24
ip rip send-packet 38.25
ip rip send version 38.26
ip rip send version 1-compatible 38.28
ip rip split-horizon 38.29
key 38.30
key chain 38.31
key-string 38.32
maximum-prefix 38.33
neighbor (RIP) 38.34
network (RIP) 38.35
offset-list (RIP) 38.36
passive-interface (RIP) 38.37
recv-buffer-size (RIP) 38.38
redistribute (RIP) 38.39
restart rip graceful 38.40
rip restart grace-period 38.41
route (RIP) 38.42
router rip 38.43
send-lifetime 38.44
show debugging rip 38.45
show ip protocols rip 38.45
show ip rip 38.46
show ip rip database 38.47
show ip rip interface 38.47
timers (RIP) 38.48
undebug rip 38.49
version 38.50

Introduction

This chapter provides an alphabetical reference of commands used to configure RIP. For more information, see **Chapter 37, RIP Configuration**.

Command List

accept-lifetime

Use this command to specify the time period during which the authentication key on a key chain is received as valid.

Use the **no** variant of this command to remove a specified time period for an authentication key on a key chain as set previously with the **accept-lifetime** command.

Syntax `accept-lifetime <start-date>{<end-date>|duration <seconds>|infinite}`

`no accept-lifetime`

Parameter	Description	
<start-date>	Specifies the start period - time and date in the format DD MMM YYYY or MMM DD YYYY: *<hh:mm:ss> {<day> <month> <year> | <month> <day> <year>}*	
	<hh:mm:ss>	Time of the day when accept-lifetime starts, in hours, minutes and seconds
	<day>	<1-31> Specifies the day of the month to start.
	<month>	Specifies the month of the year to start (the first three letters of the month, for example, Jan).
	<year>	<1993-2035> Specifies the year to start.
<end-date>	Specifies the end period - time and date in the format DD MMM YYYY or MMM DD YYYY: *<hh:mm:ss> {<day> <month> <year> | <month> <day> <year>}*	
	<hh:mm:ss>	Time of the day when lifetime expires, in hours, minutes and seconds.
	<day>	<1-31> Specifies the day of the month to expire.
	<month>	Specifies the month of the year to expire (the first three letters of the month, for example, Feb).
	<year>	<1993-2035> Specifies the year to expire.
<seconds>	<1-2147483646> Duration of the key in seconds.	
infinite	Never expires.	

Mode Keychain-key Configuration

Examples The following examples show the setting of accept-lifetime for `key1` on the key chain named `mychain`.

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)# key 1
awplus(config-keychain-key)# accept-lifetime 03:03:01 Dec 3
2007 04:04:02 Oct 6 2008
```

or:

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)# key 1
awplus(config-keychain-key)# accept-lifetime 03:03:01 3 Dec
2007 04:04:02 6 Oct 2008
```

Related Commands
key
key-string
key chain
send-lifetime

alliedware-behavior

This command configures your device to exhibit AlliedWare behavior when sending RIPv1 response/update messages. Configuring for this behavior may be necessary if you are replacing an AlliedWare device with an AlliedWare Plus device and wish to ensure consistent RIPv1 behavior.

Use the no variant of this command to implement AlliedWare Plus behavior.

This command has no impact on devices running RIPv2. Reception and transmission can be independently altered to conform to AlliedWare standard.

Syntax

```
alliedware-behavior {rip1-send|rip1-recv}
no alliedware-behavior {rip1-send|rip1-recv}
```

Parameter	Description
`rip1-send`	Configures the router to behave in AlliedWare mode when **sending** update messages.
`rip1-recv`	Configures the router to behave in AlliedWare mode when **receiving** update messages.

Default By default when sending out RIPv1 updates on an interface, if the prefix (learned through RIPv2 or otherwise redistributed into RIP) being advertised does not match the subnetting used on the outgoing RIPv1 interface it will be filtered. The **alliedware-behavior** command returns your router's RIPv1 behavior to the AlliedWare format, where the prefix will be advertised as-is.

For example, if a RIPv1 update is being sent over interface 192.168.1.4/26, by default the prefix 192.168.1.64/26 will be advertised, but the prefix 192.168.1.144/28 will be filtered because the mask /28 does not match the interface's mask of /26. If **alliedware-behavior rip1-send** is configured, the prefix 192.168.1.144 would be sent as-is.

Mode Router Configuration

Examples To configure your device for **alliedware-behavior** when sending and receiving RIPv1 update messages, enter the commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# alliedware-behavior rip1-send
awplus(config-router)# alliedware-behavior rip1-recv
```

To return your device to **AlliedWare Plus**-like behavior when sending and receiving RIPv1 update messages, enter the commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# no alliedware-behavior rip1-send
awplus(config-router)# no alliedware-behavior rip1-recv
```

Validation Commands	**show ip protocols rip** **show running-config**
Related Commands	**fullupdate (RIP)**

cisco-metric-behavior (RIP)

Use this command to enable or disable the RIP routing metric update to conform to Cisco's implementation. This command is provided to allow inter-operation with older Cisco devices that do not conform to the RFC standard for RIP route metrics.

Use the **no** variant of this command to disable this feature.

Syntax

```
cisco-metric-behavior {enable|disable}
no cisco-metric-behavior
```

Parameter	Description
`enable`	Enables updating the metric consistent with Cisco.
`disable`	Disables updating the metric consistent with Cisco.

Default By default, the Cisco metric-behavior is disabled.

Mode Router Configuration

Examples To enable the routing metric update to behave as per the Cisco implementation, enter the commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# cisco-metric-behavior enable
```

To disable the routing metric update to behave as per the default setting, enter the commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# no cisco-metric-behavior
```

Validation Commands **show running-config**

clear ip rip route

Use this command to clear specific data from the RIP routing table.

Syntax

```
clear ip rip route {<ip-dest-network/prefix-length>|static|connected|
   rip|ospf|all}
```

Parameter	Description
<ip-dest-network/prefix-length>	Removes entries which exactly match this destination address from RIP routing table. Enter the IP address and prefix length of the destination network.
`static`	Removes static entries from the RIP routing table.
`connected`	Removes entries for connected routes from the RIP routing table.
`rip`	Removes only RIP routes from the RIP routing table.
`ospf`	Removes only OSPF routes from the RIP routing table.
`all`	Clears the entire RIP routing table.

Mode Privileged Exec

Usage Using this command with the `all` parameter, clears the RIP table of all the routes.

Examples To clear the route `10.0.0.0/8` from the RIP routing table, use the following command:

```
awplus# clear ip rip route 10.0.0.0/8
```

debug rip

Use this command to specify the options for the displayed debugging information for RIP events and RIP packets.

Use the **no** variant of this command to disable the specified debug option.

Syntax

```
debug rip {events|nsm|<packet>|all}
no debug rip {events|nsm|<packet>|all}
```

Parameter	Description
`events`	RIP events debug information is displayed.
`nsm`	RIP and NSM communication is displayed.
<packet>	`packet [recv|send] [detail]` Specifies RIP packets only.
`recv`	Specifies that information for received packets be displayed.
`send`	Specifies that information for sent packets be displayed.
`detail`	Displays detailed information for the sent or received packet.
`all`	Displays all RIP debug information.

Default Disabled

Mode Privileged Exec and Global Configuration

Example The following example displays information about the RIP packets that are received and sent out from the device.

```
awplus# debug rip packet
```

Related Commands undebug rip

default-information originate (RIP)

Use this command to generate a default route into the Routing Information Protocol (RIP).

Use the **no** variant of this command to disable this feature.

Syntax

```
default-information originate
no default-information originate
```

Default Disabled

Mode Router Configuration

Usage If routes are being redistributed into RIP and the router's route table contains a default route, within one of the route categories that are being redistributed, the RIP protocol will advertise this default route, irrespective of whether the **default-information originate** command has been configured or not. However, if the router has not redistributed any default route into RIP, but you want RIP to advertise a default route anyway, then use this command.

This will cause RIP to create a default route entry in the RIP database. The entry will be of type RS (Rip Static). Unless actively filtered out, this default route will be advertised out every interface that is sending RIP. Split horizon does not apply to this route, as it is internally generated. This operates quite similarly to the OSPF **default-information originate always** command.

Example

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# default-information originate
```

default-metric (RIP)

Use this command to specify the metrics to be assigned to redistributed RIP routes.

Use the **no** variant of this command to reset the RIP metric back to its default (1).

Syntax

```
default-metric <metric>
no default-metric [<metric>]
```

Parameter	Description
<metric>	<1-16> Specifies the value of the default metric.

Default By default, the RIP metric value is set to 1.

Mode RIP Router Configuration

Usage This command is used with the **redistribute (RIP)** command to make the routing protocol use the specified metric value for all redistributed routes, regardless of the original protocol that the route has been redistributed from.

Examples This example assigns the cost of 10 to the routes that are redistributed into RIP.

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# default-metric 10
awplus(config-router)# redistribute ospf
awplus(config-router)# redistribute connected
```

Related Commands **redistribute (RIP)**

distance (RIP)

This command sets the administrative distance for RIP routes. Your device uses this value to select between two or more routes to the same destination obtained from two different routing protocols. The route with the smallest administrative distance value is added to the Forwarding Information Base (FIB). See **"Administrative Distance" on page 35.6** for more information.

The **no** variant of this command sets the administrative distance for the RIP route to the default of 120.

Syntax

```
distance <1-255> [<ip-addr/prefix-length> [<access-list>]]
no distance [<1-255>] [<ip-addr/prefix-length> [<access-list>]]
```

Parameter	Description
<1-255>	The administrative distance value you are setting for this RIP route.
<ip-addr/prefix-length>	The network IP address and prefix-length that you are changing the administrative distance for.
<access-list>	Specifies the access-list name. This access list specifies which routes within the network *<ip-address/m>* this command applies to.

Mode RIP Router Configuration

Examples To set the administrative distance to `8` for the RIP routes within the `10.0.0.0/8` network that match the access-list `mylist`, use the commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# distance 8 10.0.0.0/8 mylist
```

To set the administrative distance to the default of `120` for the RIP routes within the `10.0.0.0/8` network that match the access-list `mylist`, use the commands:

```
awplus# configure terminal
awplus (config)# router rip
awplus(config-router)# no distance 8 10.0.0.0/8 mylist
```

distribute-list (RIP)

Use this command to filter incoming or outgoing route updates using the access-list or the prefix-list.

Use the **no** variant of this command to disable this feature.

Syntax

```
distribute-list {<access-list> |prefix <prefix-list>} {in|out}
    [<interface>]

no distribute-list {<access-list> |prefix <prefix-list>} {in|out}
    [<interface>]
```

Parameter	Description
`prefix`	Filter prefixes in routing updates.
`<access-list>`	Specifies the IPv4 access-list number or name to use.
`<prefix-list>`	Specifies the name of the IPv4 prefix-list to use.
`in`	Filter incoming routing updates.
`out`	Filter outgoing routing updates.
`<interface>`	The interface on which distribute-list applies. For instance: `vlan2`

Default Disabled

Mode RIP Router Configuration

Usage Filter out incoming or outgoing route updates using access-list or prefix-list. If you do not specify the name of the interface, the filter will be applied to all interfaces.

Examples In this example the following commands are used to apply an access list called myfilter to filter incoming routing updates in `vlan2`

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# distribute-list prefix myfilter in
vlan2
```

Related Commands **access-list extended (named)**
ip prefix-list

fullupdate (RIP)

Use this command to specify which routes RIP should advertise when performing a triggered update. By default, when a triggered update is sent, RIP will only advertise those routes that have changed since the last update. When **fullupdate** is configured, the switch advertises the full RIP route table in outgoing triggered updates, including routes that have not changed. This enables faster convergence times, or allow inter-operation with legacy network equipment, but at the expense of larger update messages.

Use the **no** variant of this command to disable this feature.

Syntax

```
fullupdate
no fullupdate
```

Default By default this feature is disabled.

Mode RIP Router Configuration

Example Use the following commands to enable the fullupdate (RIP) function:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# fullupdate
```

ip rip authentication key-chain

Use this command to enable RIPv2 authentication on an interface and specify the name of the key chain to be used.

Use the **no** variant of this command to disable this function.

Syntax

```
ip rip authentication key-chain <key-chain-name>
no ip rip authentication key-chain
```

Parameter	Description
<key-chain-name>	Specify the name of the key chain. This is an alpha-numeric string, but it cannot include spaces.

Mode Interface Configuration for a VLAN interface.

Usage This command can only be used on VLAN interfaces. Use this command to perform authentication on the interface. Not configuring the key chain results in no authentication at all.

The AlliedWare Plus™ implementation provides the choice of configuring authentication for single key or multiple keys at different times. Use the **ip rip authentication string** command for single key authentication. Use the **ip rip authentication key-chain** command for multiple keys authentication. See **Chapter 37, RIP Configuration** for illustrated RIP configuration examples.

For multiple key authentication, use the following steps to configure a route to enable RIPv2 authentication using multiple keys at different times:

Step 1: Define a key chain:

In the Configure mode, identify a key chain with a key chain name using the following command:

```
awplus# configure terminal
awplus(config)# key chain <key-chain-name>
```

where *<key-chain-name>* is the name of the chain to manage, and should not include spaces.

Step 2: Define the key or keys:

In the Keychain mode, specify a key on this key chain using the following command:

```
awplus(config-keychain)# key <keyid>
```

where *<keyid>* (a decimal number in the range 1 to 2147483647) is the Key Identifier number.

Step 3: Define the authentication string or password:

In the Keychain-key mode, define the password used by a key, using the following command:

```
awplus(config-keychain-key)# key-string <key-password>
```

where *<key-password>* is a string of characters that can contain spaces, to be used as a password by the key.

Step 4: Set key management options:

This step can be performed at this stage or later when multiple keys are used. The options are configured in the keychain-key command mode.

Set the time period during which the authentication key on a key chain is received as valid, using the **accept-lifetime** command:

```
awplus(config-keychain-key)# accept-lifetime <START> <END>
```

where *<START>* and *<END>* are the beginning and end of the time period.

Set the time period during which the authentication key on a key chain can be sent, using the **send-lifetime** command:

```
awplus(config-keychain-key)# send-lifetime <START> <END>
```

where *<START>* and *<END>* are the beginning and end of the time period.

Step 5: Enable authentication on an interface:

In the Interface Configuration mode, enable authentication on the VLAN interface `vlan3` and specify the key chain to be used, using the following commands:

```
awplusawpluls# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip rip authentication key-chain <key-chain-name>
```

where *<key-chain-name>* is a set of valid authentication keys.

Step 6: Specify the mode of authentication for the given interface:

In the Interface Configuration mode for a VLAN interface, specify whether the interface uses text or MD5 authentication using:

```
awplus(config-if)# ip rip authentication mode {md5|text}
```

Example In the following sample multiple keys authentication RIP configuration, a password `toyota` is set for key 1 in key chain `cars`. Authentication is enabled on `vlan2` and the authentication mode is set to MD5:

```
awplus# configure terminal
awplus(config)# key chain cars
awplus(config-keychain)# key 1
awplus(config-keychain-key)# key-string toyota
awplus(config-keychain-key)# accept-lifetime 10:00:00 Apr 08
2008 duration 43200
awplus(config-keychain-key)# send-lifetime 10:00:00 Apr 08
2008 duration 43200
awplus(config-keychain-key)# exit
awplus(config-keychain)# exit
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication key-chain
cars
awplus(config-if)# ip rip authentication mode md5
awplus(confi-if)# exit
awplus(config)# exit
awplus#
```

Example In the following example, the VLAN interface `vlan23` is configured to use key-chain authentication with the keychain `mykey`. See the **key** command for a description of how a key chain is created.

```
awplus# configure terminal
awplus(config)# interface vlan23
awplus(config-if)# ip rip authentication key-chain mykey
```

Related Commands
accept-lifetime
send-lifetime
ip rip authentication mode
ip rip authentication string
key
key chain

ip rip authentication mode

Use this command to specify the type of authentication mode used for RIP v2 packets.

Use the **no** variant of this command to restore clear text authentication.

Syntax

```
ip rip authentication mode {md5|text}
no ip rip authentication mode
```

Parameter	Description
`md5`	Uses the keyed MD5 authentication algorithm.
`text`	Specifies clear text or simple password authentication.

Default Text authentication is enabled

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces. The AlliedWare Plus™ implementation provides the choice of configuring authentication for single key or multiple keys at different times. Use the **ip rip authentication string** command for single key authentication. Use the **ip rip authentication key-chain** command for multiple keys authentication. See **Chapter 37, RIP Configuration** for illustrated RIP configuration examples.

Use the following steps to configure a route to enable RIPv2 authentication using a single key or password:

Step 1: Define the authentication string or password

In the Interface Configuration mode for the VLAN interface `vlan2`, specify the authentication string or password used by the key, using the following command:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication string <auth-string>
```

where *<auth-string>* is the authentication string or password and it can include spaces.

Step 2: Specify the mode of authentication for the given interface:

In the Interface Configuration mode for VLAN interface `vlan2`, specify if the interface will use text or MD5 authentication, using the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication mode {md5|text}
```

See the sample below to specify `mykey` as the authentication string with MD5

authentication, for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication string mykey
awplus(config-if)# ip rip authentication mode md5
```

For multiple keys authentication, use the following steps to configure a route to enable RIPv2 authentication using multiple keys at different times:

Step 1: Define a key chain:

In the Global Configuration mode, identify a key chain with a key chain name using the following command:

```
awplus# configure terminal
awplus(config)# key chain <key-chain-name>
```

where *<key-chain-name>* is the name of the chain to manage, a text string with no spaces.

Step 2: Define the key or keys:

In the Keychain Configuration mode, specify a key on this key chain using the following command:

```
awplus(config-keychain)# key <keyid>
```

where *<keyid>* (a decimal number in the range 1 to 2147483647) is the Key Identifier number.

Step 3: Define the authentication string or password:

In the Keychain-key Configuration mode, define the password used by a key, using the following command:

```
awplus(config-keychain-key)# key-string <key-password>
```

where *<key-password>* is a string of characters that can include spaces, to be used as a password by the key.

Step 4: Set key management options:

This step can be performed at this stage or later when multiple keys are used. The options are configured in the Keychain-key Configuration mode.

Set the time period during which the authentication key on a key chain is received as valid, using the **accept-lifetime** command:

```
awplus(config-keychain-key)# accept-lifetime <start> <end>
```

where *<start>* and *<end>* are the beginning and end of the time period.

Set the time period during which the authentication key on a key chain can be sent, using

the **send-lifetime** command:

```
awplus(config-keychain-key)# send-lifetime <start> <end>
```

where <*start*> and <*end*> are the beginning and end of the time period.

Step 5: Enable authentication on an interface:

In the Interface Configuration mode, enable authentication on an interface and specify the key chain to be used, using the following command:

```
awplus(config-if)# ip rip authentication key-chain <key-
                   chain-name>
```

where <*key-chain-name*> is a set of valid authentication keys, as defined in Step 1.

Step 6: Specify the mode of authentication for the given interface:

In the Interface Configuration mode, specify whether the interface uses text or MD5 authentication using:

```
awplus(config-if)# ip rip authentication mode {md5|text}
```

Example 1 In the following sample multiple keys authentication RIP configuration, a password `toyota` is set for key 1 in key chain `cars`. Authentication is enabled on `vlan2` and the authentication mode is set to MD5:

```
                  awplus# configure terminal
            awplus(config)# key chain cars
    awplus(config-keychain)# key 1
awplus(config-keychain-key)# key-string toyota
awplus(config-keychain-key)# accept-lifetime 10:00:00 Apr 08
                             2008 duration 43200
awplus(config-keychain-key)# send-lifetime 10:00:00 Apr 08 2008
                             duration 43200
awplus(config-keychain-key)# exit
    awplus(config-keychain)# exit
            awplus(config)# interface vlan2
          awplus(config-if)# ip rip authentication key-chain
                             cars
          awplus(config-if)# ip rip authentication mode md5
          awplus(confi-if)# exit
            awplus(config)# exit
                  awplus#
```

Example 2 The following example shows md5 authentication configured on VLAN interface `vlan2`, ensuring authentication of rip packets received on this interface.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication mode md5
```

Related Commands **ip rip authentication string**
ip rip authentication key-chain

ip rip authentication string

Use this command to specify the authentication string or password used by a key.

Use the **no** variant of this command to remove the authentication string.

Syntax

```
ip rip authentication string <auth-string>
no ip rip authentication string
```

Parameter	Description
<auth-string>	The authentication string or password used by a key. It is an alpha-numeric string and can include spaces.

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces. The AlliedWare Plus™ implementation provides the choice of configuring authentication for single key or multiple keys at different times. Use this command to specify the password for a single key on an interface. Use the **ip rip authentication key-chain** command for multiple keys authentication. See **Chapter 37, RIP Configuration** for further RIP configuration examples.

Use the following steps to configure a route to enable RIPv2 authentication using a single key or password:

Step 1: Define the authentication string or password:

In the Interface Configuration mode, specify the authentication string or password used by the key, using the following commands to configure the authentication string on the VLAN interface `vlan3`:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip rip authentication string <auth-string>
```

where *<auth-string>* is the authentication string or password.

Step 2: Specify the mode of authentication for the given interface:

In the Interface Configuration mode for a VLAN, specify if the interface will use text or MD5 authentication, using the following command:

```
awplus(config-if)# ip rip authentication mode {md5|text}
```

Example See the example below to specify `mykey` as the authentication string with MD5

authentication for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication string mykey
awplus(config-if)# ip rip authentication mode md5
```

Example In the following example, the VLAN interface `vlan2` is configured to have an authentication string as `guest`. Any received RIP packet in that interface should have the same string as password.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip authentication string guest
```

Related commands **ip rip authentication key-chain**
ip rip authentication mode

ip rip receive-packet

Use this command to configure the interface to enable the reception of RIP packets.

Use the **no** variant of this command to disable this feature.

Syntax

```
ip rip receive-packet
no ip rip receive-packet
```

Default Receive-packet is enabled

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces.

Example This example shows packet receiving being turned on for the VLAN interface `vlan3`:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip rip receive-packet
```

Related Commands ip rip send-packet

ip rip receive version

Use this command to specify the version of RIP packets accepted on an interface and override the setting of the version command.

Use the **no** variant of this command to use the setting specified by the **version** command on page 38.50.

Syntax

```
ip rip receive version {[1][2]}
no ip rip receive version
```

Parameter	Description
1	Specifies acceptance of RIP version 1 packets on the interface.
2	Specifies acceptance of RIP version 2 packets on the interface.

Default Version 2

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces. This command applies to a specific VLAN interface and overrides any the version specified by the **version** command.

RIP can be run in version 1 or version 2 mode. Version 2 has more features than version 1; in particular RIP version 2 supports authentication and classless routing. Once the RIP version is set, RIP packets of that version will be received and sent on all the RIP-enabled interfaces.

Example In the following example, the VLAN interface `vlan3` is configured to receive both RIP version 1 and 2 packets:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip rip receive version 1 2
```

Related Commands **version**

ip rip send-packet

Use this command to enable sending RIP packets through the current interface.

Use the **no** variant of this command to disable this feature.

Syntax

```
ip rip send-packet
no ip rip send-packet
```

Default Send packet is enabled

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces.

Example This example shows packet sending being turned on for the VLAN interface `vlan4`:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ip rip send-packet
```

Related Commands ip rip receive-packet

ip rip send version

Use this command in Interface Configuration mode to specify the version of RIP packets sent on an interface and override the setting of the **version** command. This mechanism causes RIP version 2 interfaces to send multicast packets instead of broadcasting packets.

Use the **no** variant of this command to use the setting specified by the **version** command.

Syntax

```
ip rip send version {1|2|1 2|2 1}
no ip rip send version
```

Parameter	Description
1	Specifies the sending of RIP version 1 packets out of an interface.
2	Specifies the sending of RIP version 2 packets out of an interface.
1 2	Specifies the sending of both RIP version 1 and RIP version 2 packets out of an interface.
2 1	Specifies the sending of both RIP version 2 and RIP version 1 packets out of an interface.

Default RIP version 2 is enabled by default.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to a specific interface and overrides the version specified by the **version** command.

RIP can be run in version 1 or version 2 mode. Version 2 has more features than version 1; in particular RIP version 2 supports authentication and classless routing. Once the RIP version is set, RIP packets of that version will be received and sent on all the RIP-enabled interfaces. Selecting version parameters `1 2` or `2 1` sends RIP version 1 and 2 packets.

Use the **ip rip send version 1-compatible** command in an environment where you cannot send multicast packets. For example, in environments where multicast is not enabled and where hosts do not listen to multicast.

Examples In the following example, the VLAN interface `vlan4` is configured to send both RIP version 1 and 2 packets.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ip rip send version 1 2
```

In the following example, the VLAN interface `vlan4` is configured to send both RIP version 2 and 1 packets.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ip rip send version 2 1
```

In the following example, the VLAN interface `vlan4` is configured to send RIP version 1 packets only.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ip rip send version 1
```

In the following example, the VLAN interface `vlan4` is configured to send RIP version 2 packets only.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ip rip send version 2
```

In the following example, the VLAN interface `vlan3` is configured to use the RIP version specified by the **version** command.

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# no ip rip send version
```

Related Commands **ip rip send version 1-compatible**
version

ip rip send version 1-compatible

Use this command in Interface Configuration mode to send RIP version 1 compatible packets from a RIP version 2 interfaces to other RIP Interfaces. This mechanism causes RIP version 2 interfaces to send broadcast packets instead of multicasting packets, and is used in environments where multicast is not enabled or where hosts do not listen to multicast.

Use the **no** variant of this command to use the setting specified by the **version** command, and disable the broadcast of RIP version 2 packets that are sent as broadcast packets.

Syntax

```
ip rip send version 1-compatible
no ip rip send version
```

Parameter	Description
`1-compatible`	Specify this parameter to send RIP version 1 compatible packets from a version 2 RIP interface to other RIP interfaces. This mechanism causes version 2 RIP interfaces to broadcast packets instead of multicasting packets.

Default RIP version 2 is enabled by default.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to a specific interface and overrides the version specified by the **version** command.

RIP can be run in version 1 compatible mode. Version 2 has more features than version 1; in particular RIP version 2 supports authentication and classless routing. Once the RIP version is set, RIP packets of that version will be received and sent on all the RIP-enabled interfaces.

Use the **ip rip send version** command in an environment where you can send multicast packets. For example, in environments where multicast is enabled and where hosts listen to multicast.

Examples In the following example, the VLAN interface `vlan2` is configured to send RIP version 1-compatible packets.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip send version 1-compatible
```

In the following example, the VLAN interface `vlan3` is configured to use the RIP version specified by the **version** command.

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# no ip rip send version
```

Related Commands **ip rip send version**
version

ip rip split-horizon

Use this command to perform the split-horizon action on the interface. The default is split-horizon poisoned.

Use the **no** variant of this command to disable this function.

Syntax

```
ip rip split-horizon [poisoned]
no ip rip split-horizon
```

Parameter	Description
`poisoned`	Performs split-horizon with poisoned reverse.

Default Split horizon poisoned is the default.

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces. Use this command to avoid including routes in updates sent to the same gateway from which they were learned. Using the **split horizon** command omits routes learned from one neighbor, in updates sent to that neighbor. Using the **poisoned** parameter with this command includes such routes in updates, but sets their metrics to infinity. Thus, advertising that these routes are not reachable.

Example To perform the split-horizon action on, use the following command:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip rip split-horizon poisoned
```

key

Use this command to manage, add and delete authentication keys in a key-chain.

Use the **no** variant of this command to delete the authentication key.

Syntax

```
key <keyid>
no key <keyid>
```

Parameter	Description
<keyid>	<0-2147483647> Key identifier number.

Mode Keychain Configuration

Usage This command allows you to enter the keychain-key mode where a password can be set for the key.

Example The following example configures a key number 1 and shows the change into a **keychain-key** command mode prompt.

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)# key 1
awplus(config-keychain-key)#
```

Related Commands

key chain
key-string
accept-lifetime
send-lifetime

key chain

Use this command to enter the key chain management mode and to configure a key chain with a key chain name.

Use the **no** variant of this command to remove the key chain and all configured keys.

Syntax

```
key chain <key-chain-name>
no key chain <key-chain-name>
```

Parameter	Description
<key-chain-name>	Specify the name of the key chain to manage.

Mode Global Configuration

Usage This command allows you to enter the keychain mode from which you can specify keys on this key chain.

Example The following example shows the creation of a key chain named `mychain` and the change into **keychain** mode prompt.

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)#
```

Related Commands

key
key-string
accept-lifetime
send-lifetime

key-string

Use this command to define the password to be used by a key.

Use the **no** variant of this command to remove a password.

Syntax

```
key-string <key-password>
no key-string
```

Parameter	Description
<key-password>	A string of characters to be used as a password by the key.

Mode Keychain-key Configuration

Usage Use this command to specify passwords for different keys.

Examples In the following example, the password for `key1` in the key chain named `mychain` is set to password **prime**:

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)# key 1
awplus(config-keychain-key)# key-string prime
```

In the following example, the password for `key1` in the key chain named `mychain` is removed:

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)# key 1
awplus(config-keychain-key)# no key-string
```

Related Commands

key
key chain
accept-lifetime
send-lifetime

maximum-prefix

Use this command to configure the maximum number of RIP routes stored in the routing table.

Use the **no** variant of this command to disable all limiting of the number of RIP routes stored in the routing table.

Syntax

```
maximum-prefix <maxprefix> [<threshold>]
no maximum-prefix
```

Parameter	Description
<maxprefix>	<1-65535> The maximum number of RIP routes allowed.
<threshold>	<1-100> Percentage of maximum routes to generate a warning. The default threshold is 75%.

Mode Router Configuration

Example To configure the maximum number of RIP routes to 150, use the following command:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# maximum-prefix 150
```

neighbor (RIP)

Use this command to specify a neighbor router. It is used for each router to which you wish to send unicast RIP updates.

Use the **no** variant of this command to stop sending unicast updates to the specific router.

Syntax

```
neighbor <ip-address>
no neighbor <ip-address>
```

Parameter	Description
<ip-address>	The IP address of a neighboring router with which the routing information will be exchanged.

Default Disabled

Mode Router Configuration

Usage Use this command to exchange nonbroadcast routing information. It can be used multiple times for additional neighbors.

The **passive-interface (RIP)** command disables sending routing updates on an interface. Use the `neighbor` command in conjunction with the **passive-interface (RIP)** to send routing updates to specific neighbors.

Example To specify the neighbor router to 1.1.1.1, use the following command:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# passive-interface vlan1
awplus(config-router)# neighbor 1.1.1.1
```

Related Commands **passive-interface (RIP)**

network (RIP)

Use this command to activate the transmission of RIP routing information on the defined network.

Use the **no** variant of this command to remove the specified network or VLAN as one that runs RIP.

Syntax

```
network {<network-address>[/<subnet-prefix-length>]|<vlan-name>}
no network {<network-address>[/<subnet-mask>]|<vlan-name>}
```

Parameter	Description
<network-address> *[/<subnet-prefix-length>]*	Specifies the network address to run RIP. Entering a subnet mask (or prefix length) for the network address is optional. Where no mask is entered, the switch will attempt to apply a mask that is appropriate to the class (A, B, or C) of the address entered, i.e. an IP address of 10.0.0.0 will have a prefix length of 8 applied to it.
<vlan-name>	Specify a VLAN name with up to 32 alphanumeric characters to run RIP.

Default Disabled

Mode RIP Router Configuration

Usage Use this command to specify networks, or VLANs, to which routing updates will be sent and received. The connected routes corresponding to the specified network, or VLANs, will be automatically advertised in RIP updates. RIP updates will be sent and received within the specified network or VLAN.

Example Use the following commands to activate RIP routing updates on network 172.16.20.0/24:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# network 172.16.20.0/24
```

Related Commands

show ip rip
show running-config
clear ip rip route

offset-list (RIP)

Use this command to add an offset to the **in** and **out** metrics of routes learned through RIP.

Use the **no** variant of this command to remove the offset list.

Syntax

```
offset-list <access-list> {in|out} <offset> [<interface>]
no offset-list <access-list> {in|out} <offset> [<interface>]
```

Parameter	Description
<access-list>	Specifies the access-list number or names to apply.
`in`	Indicates the access list will be used for metrics of incoming advertised routes.
`out`	Indicates the access list will be used for metrics of outgoing advertised routes.
<offset>	`<0-16>` Specifies that the offset is used for metrics of networks matching the access list.
<interface>	An alphanumeric string that specifies the interface to match.

Default The default `offset value` is the metric value of the interface over which the updates are being exchanged.

Mode RIP Router Configuration

Usage Use this command to specify the offset value that is added to the routing metric. When the networks match the access list the offset is applied to the metrics. No change occurs if the offset value is zero.

Examples In this example the router examines the RIP updates being sent out from interface `vlan2` and adds `5` hops to the routes matching the ip addresses specified in the access list `8`.

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# offset-list 8 in 5 vlan2
```

Related Commands **access-list (extended numbered)**

passive-interface (RIP)

Use this command to block RIP broadcasts on the VLAN interface.

Use the **no** variant of this command to disable this function.

Syntax

```
passive-interface <interface>
no passive-interface <interface>
```

Parameter	Description
<interface>	Specifies the interface name.

Default Disabled

Mode RIP Router Configuration

Usage This command can only be configured for VLAN interfaces.

Examples Use the following commands to block RIP broadcasts on vlan20:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# passive-interface vlan20
```

Related Commands show ip rip

recv-buffer-size (RIP)

Use this command to run-time configure the RIP UDP (User Datagram Protocol) receive-buffer size to improve UDP reliability by avoiding UDP receive buffer overrun.

Use the **no** variant of this command to reset the configured RIP UDP receive-buffer size to the system default (196608 bits).

Syntax

```
recv-buffer-size <8192-2147483647>

no recv-buffer-size [<8192-2147483647>]
```

Parameter	Description
<8192-2147483647>	Specify the RIP UDP (User Datagram Protocol) buffer size value in bits.

Default 196608 bits is the system default when reset using the **no** variant of this command.

Mode Router Configuration

Examples To run-time configure the RIP UDP, use the following commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# recv-buffer-size 23456789

awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# no recv-buffer-size 23456789
```

redistribute (RIP)

Use this command to redistribute information from other routing protocols into RIP.

Use the **no** variant of this command to disable the specified redistribution. The parameters **metric** and **routemap** may be used on this command, but have no effect.

Syntax

```
redistribute {connected|static|ospf} [metric <0-16>] [routemap
   <routemap>]
no redistribute {connected|static|ospf} [metric] [routemap]
```

Parameter	Description
`routemap`	Optional. Specifies route-map that controls how routes are redistributed.
`<routemap>`	Optional. The name of the route map.
`connected`	Redistribute from connected routes.
`static`	Redistribute from static routes.
`ospf`	Redistribute from Open Shortest Path First (OSPF).
`metric <0-16>`	Optional. Sets the value of the metric that will be applied to routes redistributed into RIP from other protocols. If a value is not specified, and no value is specified using the **default-metric (RIP)** command, the default is one.

Default By default, the RIP metric value is set to 1.

Mode RIP Router Configuration

Example To apply the metric value 15 to static routes being redistributed into RIP, use the commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# redistribute static metric 15
```

Related Commands **default-metric (RIP)**

restart rip graceful

Use this command to force the RIP process to restart, and optionally set the grace-period.

Syntax `restart rip graceful [grace-period <1-65535>]`

Mode Privileged Exec

Default The default RIP grace-period is 60 seconds.

Usage After this command is executed, the RIP process immediately shuts down. It notifies the system that RIP has performed a graceful shutdown. Routes that have been installed into the route table by RIP are preserved until the specified grace-period expires.

When a **restart rip graceful** command is issued, the RIP configuration is reloaded from the last saved configuration. Ensure you first enter the command **copy running-config startup-config**.

When a master failover happens on a VCStack, the RIP grace-period will apply the larger value of either, the setting's configured value, or its default of 60 seconds.

Example To apply a restart rip graceful setting, grace-period to 100 seconds use the following commands:

```
awplus# copy running-config startup-config
awplus# restart rip graceful grace-period 100
```

rip restart grace-period

Use this command to change the grace period of RIP graceful restart.

Use the **no** variant of this command to disable this function.

Syntax

```
rip restart grace-period <1-65535>
no rip restart grace-period <1-65535>
```

Mode Global Configuration

Default The default RIP grace-period is 60 seconds.

Usage Use this command to enable the **Graceful Restart** feature on the RIP process. Entering this command configures a grace period for RIP.

When a master failover happens on a VCStack, the RIP grace-period will be the longest period between the default value (60 seconds is the default RIP grace-period) and the configured RIP grace-period value from this command. So the configured RIP grace-period value will not be used for a VCStack master failover if it is shorter than the default RIP grace-period value.

Example

```
awplus# configure terminal
awplus(config)# rip restart grace-period 200
```

route (RIP)

Use this command to configure static RIP routes.

Use the **no** variant of this command to disable this function.

Syntax

```
route <ip-addr/prefix-length>
no route <ip-addr/prefix-length>
```

Parameter	Description
<ip-addr/prefix-length>	The IPv4 address and prefix length.

Default No static RIP route is added by default.

Mode RIP Router Configuration

Usage Use this command to add a static RIP route. After adding the RIP route, the route can be checked in the RIP routing table.

Example To create a static RIP route to IP subnet `192.168.1.0/24`, use the following commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# route 192.168.1.0/24
```

Related Commands

show ip rip
clear ip rip route

router rip

Use this global command to enter Router Configuration mode to enable the RIP routing process.

Use the **no** variant of this command to disable the RIP routing process.

Syntax

```
router rip
no router rip
```

Mode Global Configuration

Example This command is used to begin the RIP routing process:

```
                awplus# configure terminal
        awplus(config)# router rip
  awplus(config-router)# version 1
  awplus(config-router)# network 10.10.10.0/24
  awplus(config-router)# network 10.10.11.0/24
  awplus(config-router)# neighbor 10.10.10.10
```

Related Commands network (RIP)
version

send-lifetime

Use this command to specify the time period during which the authentication key on a key chain can be sent.

Syntax

```
send-lifetime <start-date>{<end-date>|duration <seconds>|infinite}
no send-lifetime
```

Parameter	Description	
<start-date>	Specifies the start period - time and date in the format DD MMM YYYY or MMM DD YYYY: *<hh:mm:ss> {<day> <month> <year> | <month> <day> <year>}*	
	<hh:mm:ss>	Time of the day when send-lifetime starts, in hours, minutes and seconds
	<day>	<1-31> Specifies the day of the month to start.
	<month>	Specifies the month of the year to start (the first three letters of the month, for example, Jan).
	<year>	<1993-2035> Specifies the year to start.
<end-date>	Specifies the end period - time and date in the format DD MMM YYYY or MMM DD YYYY: *<hh:mm:ss> {<day> <month> <year> | <month> <day> <year>}*	
	<hh:mm:ss>	Time of the day when lifetime expires, in hours, minutes and seconds.
	<day>	<1-31> Specifies the day of the month to expire.
	<month>	Specifies the month of the year to expire (the first three letters of the month, for example, Feb).
	<year>	<1993-2035> Specifies the year to expire.
<seconds>	<1-2147483646> Duration of the key in seconds.	
infinite	Never expires.	

Mode Keychain-key Configuration

Example The following example shows the setting of send-lifetime for key1 on the key chain named mychain.

```
awplus# configure terminal
awplus(config)# key chain mychain
awplus(config-keychain)# key 1
awplus(config-keychain-key)# send-lifetime 03:03:01 Jan 3 2004
04:04:02 Dec 6 2006
```

Related Commands

key
key-string
key chain
accept-lifetime

show debugging rip

Use this command to display the RIP debugging status for these debugging options: nsm debugging, RIP event debugging, RIP packet debugging and RIP nsm debugging.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax

```
show debugging rip
```

Mode

User Exec and Privileged Exec

Usage

Use this command to display the debug status of RIP.

Example

```
awplus# show debugging rip
```

show ip protocols rip

Use this command to display RIP process parameters and statistics.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax

```
show ip protocols rip
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show ip protocols rip
```

Output

Figure 38-1: Example output from the show ip protocols rip command

```
Routing Protocol is "rip"
Sending updates every 30 seconds with +/-50%, next due in 12
seconds
Timeout after 180 seconds, garbage collect after 120 seconds
Outgoing update filter list for all interface is not set
Incoming update filter list for all interface is not set
Default redistribution metric is 1
Redistributing: connected static
Default version control: send version 2, receive version 2
Interface        Send  Recv   Key-chain
   vlan25          2     2
Routing for Networks:
  10.10.0.0/24
Routing Information Sources:
  Gateway          BadPackets BadRoutes  Distance Last Update
Distance: (default is 120
```

show ip rip

Use this command to show RIP routes.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip rip`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip rip
```

Output **Figure 38-2: Example output from the show up rip command**

```
awplus#show ip rip
Codes: R - RIP, Rc - RIP connected, Rs - RIP static
       C - Connected, S - Static, O - OSPF
Network        Next Hop Metric From If   Time
C 10.0.1.0/24             1      vlan20
S 10.10.10.0/24           1      vlan20
C 10.10.11.0/24           1      vlan20
S 192.168.101.0/24        1      vlan20
R 192.192.192.0/24        1      --
```

Related Commands

route (RIP)
network (RIP)
clear ip rip route

show ip rip database

Use this command to display information about the RIP database.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip rip database [full]`

Parameter	Description
`full`	Specify the full RIP database including sub-optimal RIP routes.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip rip database
awplus# show ip rip database full
```

Related Commands show ip rip

show ip rip interface

Use this command to display information about the RIP interfaces. You can specify an interface name to display information about a specific interface.

Syntax `show ip rip interface [<interface>]`

Parameter	Description
<interface>	The interface to display information about. For instance: `vlan2`.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip rip interface
```

timers (RIP)

Use this command to adjust routing network timers.

Use the **no** variant of this command to restore the defaults.

Syntax

```
timers basic <update> <timeout> <garbage>
no timers basic
```

Parameter	Description
<update>	<5-2147483647> Specifies the period at which RIP route update packets are transmitted. The default is 30 seconds.
<timeout>	<5-2147483647> Specifies the routing information timeout timer in seconds. The default is 180 seconds. After this interval has elapsed and no updates for a route are received, the route is declared invalid.
<garbage>	<5-2147483647> Specifies the routing garbage collection timer in seconds. The default is 120 seconds.

Default Enabled

Mode RIP Router Configuration

Usage This command adjusts the RIP timing parameters.

The update timer is the time between sending out updates, that contain the complete routing table, to every neighboring router.

If an update for a given route has not been seen for the time specified by the timeout parameter, that route is no longer valid. However, it is retained in the routing table for a short time, with metric 16, so that neighbors are notified that the route has been dropped.

When the time specified by the garbage parameter expires the metric 16 route is finally removed from the routing table. Until the garbage time expires, the route is included in all updates sent by the router.

All the routers in the network must have the same timers to ensure the smooth operation of RIP throughout the network.

Examples To adjust router network timers to 30 180 120, use the following command:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# timers basic 30 180 120
```

To adjust router network timers to 30 180 120 with VRF, use the following command:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# address-family ipv4 vrf blue
awplus(config-router-af)# timers basic 30 180 120
```

undebug rip

Use this command to disable the options set for debugging information of RIP events, packets and communication between RIP and NSM.

This command has the same effect as the **no debug rip** command.

Syntax `undebug rip {all|events|nsm|<packet>}`

Parameter	Description
`all`	Disables all RIP debugging.
`events`	Disables the logging of RIP events.
`nsm`	Disables the logging of RIP and NSM communication.
<packet>	`packet [recv|send] [detail]` Disables the debugging of RIP packets.
`recv`	Disables the logging of received packet information.
`send`	Disables the logging of sent packet information.
`detail`	Disables the logging of sent or received RIP packets.

Mode Privileged Exec

Example To disable the options set for debugging RIP information events, use the following command:

```
awplus# undebug rip packet
```

Related Commands debug rip

version

Use this command to specify a RIP version used globally by the router.

Use the **no** variant of this command to restore the default version.

Syntax

```
version {1|2}
no version
```

Parameter	Description
1|2	Specifies the version of RIP processing.

Default Version 2

Mode RIP Router Configuration

Usage RIP can be run in version 1 or version 2 mode. Version 2 has more features than version 1; in particular RIP version 2 supports authentication and classless routing. Once the RIP version is set, RIP packets of that version will be received and sent on all the RIP-enabled interfaces.

Setting the version command has no impact on receiving updates, only on sending them. The **ip rip send version** command overrides the value set by the **version** command on an interface-specific basis. The **ip rip receive version** command allows you to configure a specific interface to accept only packets of the specified RIP version. The **ip rip receive version** command and the **ip rip send version** command override the value set by this command.

Examples To specify a RIP version, use the following commands:

```
awplus# configure terminal
awplus(config)# router rip
awplus(config-router)# version 1
```

Validation Commands **show running-config**

```
awplus#show running-config
!
router rip
 version 1
!
```

Related Commands **ip rip receive version**
ip rip send version

Chapter 39: RIPng for IPv6 Configuration

Introduction 39.2
Enabling RIPng 39.2
Troubleshooting RIPng Adjacency 39.5

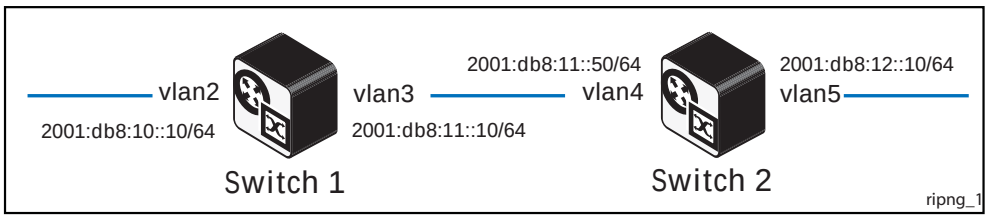
Introduction

This chapter contains a basic RIPng configuration example. To see details on the RIPng commands used in these examples, or to see the outputs of the Validation commands, refer to **Chapter 40, RIPng for IPv6 Commands**.

Enabling RIPng

This example shows the minimum configuration required for enabling RIPng on an interface. Switch 1 and Switch 2 are two switches connected via network **2001:db8:11::/64**.

To enable RIPng, first add IPv6 addresses to interfaces and then enable RIPng on each interface.



Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ipv6 rip`	Globally enable RIPng routing on `Switch 1` and enter Router Configuration mode.
`awplus(config-router)#` `exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ipv6 address 2001:db8:10::10/64`	Configure the IPv6 address on interface `vlan2`.
`awplus(config-if)#` `ipv6 router rip`	Enable RIPng routing on interface `vlan2`.
`awplus(config-if)#` `exit`	Exit Interface mode and enter Global Configuration mode.
`awplus(config)#` `interface vlan3`	Specify the interface (`vlan3`) and enter the Interface Configuration mode.

Switch 1(cont.)

`awplus(config-if)#` `ipv6 address 2001:db8:11::10/64`	Configure the IPv6 address on interface `vlan3`.
`awplus(config-if)#` `ipv6 router rip`	Enable RIPng routing on interface `vlan3`.
`awplus(config-if)#` `exit`	Exit Interface mode and enter Global Configuration mode.
`awplus(config)#` `router ipv6 rip`	Define a RIPng routing process and enter Router Configuration mode.
`awplus(config-router)#` `exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#` `exit`	Exit Global Configuration mode and return to Privileged Exec mode.

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ipv6 rip`	Globally enable RIPng routing on `Switch 2` and enter Router Configuration mode.
`awplus(config-router)#` `exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#` `interface vlan5`	Specify the interface (`vlan5`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ipv6 address 2001:db8:12::10/64`	Configure the IPv6 address on interface `vlan5`.
`awplus(config-if)#` `ipv6 router rip`	Enable RIPng routing on interface `vlan5`
`awplus(config-if)#` `exit`	Exit Interface mode and enter Global Configuration mode.
`awplus(config)#` `interface vlan4`	Specify the interface (`vlan4`) and enter the Interface Configuration mode.

Switch 2 (cont.)

`awplus(config-if)#` `ipv6 address 2001:db8:11::50/64`	Configure the IPv6 address on interface `vlan4`.
`awplus(config-if)#` `ipv6 router rip`	Enable RIPng routing on interface `vlan4`.
`awplus(config-if)#` `exit`	Exit Interface mode and enter Global Configuration mode.
`awplus(config)#` `router ipv6 rip`	Define a RIPng routing process and enter Router Configuration mode.
`awplus(config-router)#` `exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#` `exit`	Exit Global Configuration mode and return to Privileged Exec mode.

Names of commands used

router ipv6 rip
ipv6 router rip

Validation commands

show ipv6 rip

Troubleshooting RIPng Adjacency

Follow the steps below to troubleshoot RIPng adjacency:

Step 1. Confirm the interface is not shutdown

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router ipv6 rip`	Globally enable RIPng routing on the Switch and enter Router Configuration mode.
`awplus(config-router)#`	
`exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#`	
`interface vlan2`	Specify the interface (`vlan2`) and enter the Interface mode.
`awplus(config-if)#`	
`no shutdown`	Ensure the interface is not administratively shutdown.
`awplus(config-if)#`	
`exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#`	
`exit`	Exit Global Configuration mode and return to Privileged Exec mode.
`awplus#`	
`show ipv6 interface brief`	Use the `show interface` command to make sure that the interface is not administratively shutdown.

Step 2. Confirm RIPng is enabled on the interface

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router ipv6 rip`	Globally enable RIPng routing on the Switch.
`awplus(config-router)#`	
`exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#`	
`interface vlan2`	Specify the interface (`vlan2`) and enter the Interface Configuration mode.

Step 2. Confirm RIPng is enabled on the interface(cont.)

`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) and enter the Interface Configuration mode.
`awplus(config-if)#` `ipv6 router rip`	Enable RIPng routing on interface `vlan2`.
`awplus(config-if)#` `exit`	Exit Interface Configuration mode and enter Global Configuration mode.
`awplus(config)#` `router ipv6 rip`	Define a RIPng routing process and enter Router Configuration mode.
`awplus(config-router)#` `exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#` `exit`	Exit Global Configuration mode and return to Privileged Exec mode.

`awplus#`
`show ipv6 rip interface vlan2`
	Check the configuration. E.g.:
	`vlan2 is up, line protocol is up`
	`  Routing Protocol: RIPng`
	`  Passive interface: Disabled`
	`  Split horizon: Enabled with Poisoned Reversed`
	`  IPv6 interface address:`
	`  2001:db8:1::10/64`
	`  2001:db8:204:76ff:fec8::/10`

Step 3. Check the interface is not a passive interface

`awplus#`	

`show running-config`	Check that the interface is not configured as a passive interface.
	If it is configured to be passive, this is displayed in the output from the **show running config** command, e.g.:
	`!`
	`router ipv6 rip`
	`passive interface vlan2`
	`!`

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router ipv6 rip`	Enter the Router Configuration mode.
`awplus(config-router)#`	
`no passive interface port1.0.1`	Remove the passive interface.

Step 4. Ensure RIPng advertisements get exchanged

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router ipv6 rip`	Globally enable RIPng routing on the Switch and enter Router Configuration mode.
`awplus(config-router)#`	
`exit`	Exit Router Configuration mode and return to Global Configuration mode.
`awplus(config)#`	
`debug ipv6 rip event`	Check on the interface to make sure that RIPng advertisements are being sent and received.
`awplus(config)#`	
`debug ipv6 rip packet detail`	Check on the interface to make sure that RIPng advertisements are being sent and received.
`awplus(config)#`	
`exit`	Exit Global Configuration mode to enter Privileged Exec mode.
`awplus#`	
`terminal monitor`	View log messages, or use a packet sniffer such as Ethereal or Wireshark, to verify RIPng advertisements.

Chapter 40: RIPng for IPv6 Commands

Introduction 40.2
Command List 40.2
aggregate-address (IPv6 RIPng) 40.3
clear ipv6 rip route 40.4
debug ipv6 rip 40.5
default-information originate (IPv6 RIPng) 40.6
default-metric (IPv6 RIPng) 40.7
distribute-list (IPv6 RIPng) 40.8
ipv6 rip metric-offset 40.9
ipv6 rip split-horizon 40.11
ipv6 router rip 40.12
neighbor (IPv6 RIPng) 40.13
offset-list (IPv6 RIPng) 40.14
passive-interface (IPv6 RIPng) 40.15
recv-buffer-size (IPv6 RIPng) 40.16
redistribute (IPv6 RIPng) 40.17
route (IPv6 RIPng) 40.18
router ipv6 rip 40.18
show debugging ipv6 rip 40.19
show ipv6 protocols rip 40.19
show ipv6 rip 40.20
show ipv6 rip database 40.21
show ipv6 rip interface 40.22
timers (IPv6 RIPng) 40.23
undebug ipv6 rip 40.24

Introduction

This chapter contains RIPng commands. RIPng (Routing Information Protocol next generation) is an extension of RIPv2 to support IPv6. RFC 2080 specifies RIPng. The differences between RIPv2 and RIPng are:

- RIPng does not support RIP updates authentication
- RIPng does not allow the attachment of arbitrary tags to routes
- RIPng requires the encoding of the next-hop for a set of routes

For more information, see **Chapter 39, RIPng for IPv6 Configuration**.

Command List

This section provides an alphabetical reference of commands used to configure RIPng for IPv6.

aggregate-address (IPv6 RIPng)

Use this command to add an aggregate route to RIPng.

Use the **no** variant of this command to remove the aggregate route from RIPng.

Syntax

```
aggregate-address <ipv6-addr/prefix-length>
no aggregate-address <ipv6-addr/prefix-length>
```

Parameter	Description
<ipv6-addr/prefix-length>	Specify the IPv6 Address in the format *X:X::X:X/Prefix-Length*. The prefix-length is a decimal integer between 1 and 128.

Mode Router Configuration

Usage The route will not be added to the RIPng database unless the database contains at least one route which is contained within the address range covered by the aggregate route. As soon as there are any such component routes in the RIPng database, then the following occurs:

- the aggregate route is added to the RIPng database
- all the component routes that are within the address range covered by the aggregate route are retained in the RIPng database, but are marked as suppressed routes. The aggregate route will be advertised in RIPng updates, and the component route will no longer be advertised.

Note that simply having a component route in the IPv6 route database is not a sufficient condition for the aggregate route to be included into the RIPng database. The component route(s) must be in the RIPng database before the aggregate route will be included in the RIPng database. There is no restriction on the method by which the component routes have arrived into the RIPng database, it can be by being connected RIP interfaces, by redistribution or by direct inclusion using the **route** command in router IPv6 RIP configuration mode.

Example

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# aggregate-address 2001:db8::/32
```

clear ipv6 rip route

Use this command to clear specific data from the RIPng routing table.

Syntax

```
clear ipv6 rip route {<ipv6-addr/prefix-length>|all|connected|rip|
   static|ospf}
```

Parameter	Description
<ipv6-addr/prefix-length>	Specify the IPv6 Address in format *X:X::X:X/Prefix-Length*. The prefix-length is a decimal integer between 1 and 128. Removes entries which exactly match this destination address from the RIPng routing table.
`connected`	Removes redistributed connected entries from RIPng routing table.
`static`	Removes redistributed static entries from the RIPng routing table.
`rip`	Removes RIPng routes from the RIPng routing table.
`ospf`	Removes redistributed OSPFv3 routes from the RIPng routing table.
`all`	Clears the entire RIPng routing table.

Mode Privileged Exec

Example

```
awplus# clear ipv6 rip route all
awplus# clear ipv6 rip route 2001:db8::/32
```

debug ipv6 rip

Use this command to enable RIPng debugging and specify debugging for RIPng events, RIPng packets, or RIPng communication with NSM processes.

Use the **no** variant of this command to disable RIPng debugging.

Syntax

```
debug ipv6 rip [all|events|nsm|packet [detail]|recv [detail]|
    send [detail]]

no debug ipv6 rip [all|events|nsm|packet [detail]|recv [detail]|
    send [detail]]
```

Parameter	Description
`all`	Displays all RIPng debugging showing RIPng events debug information, RIPng received packets information, and RIPng sent packets information.
`events`	Displays RIPng events debug information.
`nsm`	Displays RIPng and NSM communication.
`packet`	Displays RIPng packets only.
`recv`	Displays information for received packets.
`send`	Displays information for sent packets.
`detail`	Displays detailed information for the sent or received packet.

Default RIPng debugging is disabled by default.

Mode Privileged Exec and Global Configuration

Example

```
awplus# debug ipv6 rip events
awplus# debug ipv6 rip packet send detail
awplus# debug ipv6 rip nsm
```

Related Commands undebug ipv6 rip

default-information originate (IPv6 RIPng)

Use this command to generate a default route into RIPng.

Use the **no** variant of this command to disable this feature.

Syntax

```
default-information originate
no default-information originate
```

Default Disabled

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# default-information originate
```

default-metric (IPv6 RIPng)

Use this command to specify the metrics to be assigned to redistributed RIPng routes.

Use the **no** variant of this command to reset the RIPng metric back to its default (1).

Syntax

```
default-metric <1-16>
no default-metric [<1-16>]
```

Parameter	Description
<1-16>	Metric value.

Default By default, the RIPng metric value is set to 1.

Mode Router Configuration

Usage This command is used with the **redistribute (IPv6 RIPng)** command to make the routing protocol use the specified metric value for all redistributed RIPng routes, regardless of the original protocol that the route has been redistributed from.

Note, this metric is not applied to routes that are brought into RIPng by using the **route** command in router IPv6 RIP configuration mode. This metric is, though, applied to any RIPng aggregate routes that have been brought into the RIPng database due to the presence of a component route that was redistributed into RIPng.

Also note that the default-metric is applied to routes redistributed into RIPng with no metric assignment in the routemap associated with redistribution.

Example

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# default-metric 8
```

Related Commands **ipv6 rip metric-offset**
redistribute (IPv6 RIPng)

distribute-list (IPv6 RIPng)

Use this command to filter incoming or outgoing route updates using the access-list or the prefix-list.

Use the **no** variant of this command to disable this feature.

Syntax

```
distribute-list [<access-list>|prefix <prefix-list-name>] [in|out]
    [<interface>]
no distribute-list [<access-list>|prefix <prefix-list-name>] [in|out]
    [<interface>]
```

Parameter	Description
<access-list>	Specifies the IPv6 access-list number or name to use.
<prefix-list-name>	Filter prefixes in routing updates. Specify the name of the IPv6 prefix-list to use.
<interface>	The interface for which distribute-list applies. For instance: `vlan2`.
`in`	Filter incoming routing updates.
`out`	Filter outgoing routing updates.

Default Disabled

Mode Router Configuration

Usage Filter out incoming or outgoing route updates using the access-list or the prefix-list. If you do not specify the name of the interface, the filter is applied to all the interfaces.

Example To filter incoming or outgoing route updates, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# distribute-list prefix myfilter in vlan2
```

Related Commands **ipv6 access-list extended (named)**
ipv6 nd prefix

ipv6 rip metric-offset

Use this command to increment the metric value on incoming routes for a specified interface. This command can be used to artificially inflate the metric value for routes learned on the specified interface. Routes learned on the specified interface are only used if the routes to the same destination with a lower metric value in the routing table are down.

Use the **no** variant of this command to reset the metric value on incoming routes to the default value (1). You can set the metric value for redistributed routes with **default-metric (IPv6 RIPng)** and **redistribute (IPv6 RIPng)** commands in Router Configuration mode.

Syntax

```
ipv6 rip metric-offset <1-16>

no ipv6 rip metric-offset <1-16>
```

Parameter	Description
<1-16>	Specify an increment to the metric value on an incoming route. The metric value for RIPng routes is the hop count for the route.

Default The default RIPng metric value is 1.

Usage When a RIPng route is received on a VLAN interface, the metric value for the interface set by this command is added to the metric value of the route in the routing table. Note this command only increments the metric for incoming routes on a specified interface. Increasing the metric value for a VLAN interface increases the metric value of routes received on that VLAN interface. This changes the route selected from the routing table.

The RIPng metric is the hop count. At regular intervals of the routing update timer (which has a default value of 30 seconds), and at the time of change in the topology, the RIPng router sends update messages to other routers. The listening routers update their route table with the new route, and increase the metric value of the path by one (referred to as a hop count). The router recognizes the IPv6 address advertising router as the next hop, then sends the routing updates to other routers. A maximum allowable hop count is 15. If a router reaches a metric value of 16 or more, the destination is identified as unreachable.

See how AlliedWare Plus adds routes in the **Chapter 35, How AlliedWare Plus Adds Routes** in **Chapter 35, Route Selection**. See also the **default-metric (IPv6 RIPng)** and **redistribute (IPv6 RIPng)** commands to specify the metric for redistributed RIPng routes.

Examples To increment the metric-offset on the VLAN interface vlan2, enter the below commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# exit
awplus(config)# interface vlan2
awplus(config-if)# ipv6 rip metric-offset 1
```

To reset the metric-offset on the VLAN interface `vlan2` to the default value, enter the below commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 rip metric-offset 1
```

Validation Commands **show running-config**

Related Commands **default-metric (IPv6 RIPng)**

ipv6 rip split-horizon

Use this command to perform the split-horizon action on the interface. The default is split-horizon with poisoned reverse.

Use the **no** variant of this command to disable this function.

Syntax

```
ipv6 rip split-horizon [poisoned]
no ipv6 rip split-horizon
```

Parameter	Description
`split-horizon`	Perform split-horizon without poisoned reverse
`poisoned`	Performs split-horizon with poisoned reverse.

Default Split-horizon with poisoned reverse is the default.

Usage Use this command to avoid including routes in updates sent to the same gateway from which they were learned. Using the **split horizon** command omits routes learned from one neighbor, in updates sent to that neighbor. Using the **poisoned** parameter with this command includes such routes in updates, but sets their metrics to infinity. Thus, advertising that these routes are not reachable.

Examples To perform split-horizon with poisoned reverse on the VLAN interface `vlan2`, enter the below commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# exit
awplus(config)# interface vlan2
awplus(config-if)# ipv6 rip split-horizon poisoned
```

To disable split-horizon on the VLAN interface `vlan2`, enter the below commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 rip split-horizon
```

Validation Commands **show running-config**

ipv6 router rip

Use this command to enable RIPng routing on an interface.

Use the **no** variant of this command to disable RIPng routing on an interface.

Syntax

```
ipv6 router rip
no ipv6 router rip
```

Default RIPng routing is disabled by default.

Usage This command can only be configured on VLAN interfaces.

Examples To enable RIPng routing on the VLAN interface `vlan2`, enter the below commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# exit
awplus(config)# interface vlan2
awplus(config-if)# ipv6 router rip
```

To disable RIPng routing on the VLAN interface `vlan2`, enter the below commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 router rip
```

neighbor (IPv6 RIPng)

Use this command to specify a neighbor router.

Use the **no** variant of this command to disable the specific router.

Syntax

```
neighbor <ipv6-link-local-addr> <interface>
no neighbor <ipv6-link-local-addr> <interface>
```

Parameter	Description
<ipv6-link-local-addr>	Specify the link-local IPv6 address (in the format *X:X::X:X*) of a neighboring router to exchange routing information with.
<interface>	The interface. For instance: `vlan2`.

Mode Router Configuration

Usage Use this command to exchange non broadcast routing information. It can be used multiple times for additional neighbors.

The **passive-interface (IPv6 RIPng)** command disables sending routing updates on an interface. Use the `neighbor` command in conjunction with the **passive-interface (IPv6 RIPng)** command to send routing updates to specific neighbors.

Examples

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# neighbor 2001:db8:1::1 vlan2

awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# no neighbor 2001:db8:1::1 vlan2
```

Related Commands **passive-interface (IPv6 RIPng)**

offset-list (IPv6 RIPng)

Use this command to add an offset to in and out metrics to routes learned through RIPng.

Use the **no** variant of this command to remove an offset list.

Syntax

```
offset-list {<access-list-number>|<access-list-name>} {in|out}
    <offset> [<interface>]

no offset-list {<access-list-number>|<access-list-name>} {in|out}
    <offset> [<interface>]
```

Parameter	Description
<access-list-number>	Specify an access-list number to apply to an offset-list.
<access-list-name>	Specify and access-list name to apply to an offset-list.
`in`	Indicates the access-list will be used for metrics of incoming advertised routes
`out`	Indicates the access-list will be used for metrics of outgoing advertised routes
<offset>	`<0-16>` Specifies that the offset is used for metrics of networks matching the access-list
<interface>	The interface to match. For instance: `vlan2`.

Default The default offset value is the metric value of the interface over which the updates are being exchanged.

Mode Router Configuration

Usage Use this command to specify the offset value that is added to the routing metric. When the networks match the access list the offset is applied to the metrics. No change occurs if the offset value is zero.

Example In this example the router examines the RIPng updates being sent out from interface `vlan2` and adds `8` hops to the routes matching the ip addresses specified in the access list `2`.

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# offset-list mylist in 8 vlan2
```

passive-interface (IPv6 RIPng)

Use this command to enable suppression of routing updates on an interface.

Use the **no** variant of this command to disable this function.

Syntax

```
passive-interface <interface>
no passive-interface <interface>
```

Parameter	Description
<interface>	The interface. For instance: `vlan2`.

Default Disabled

Mode Router Configuration

Examples To enable suppression of routing updates, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# passive-interface vlan2

awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# no passive-interface vlan2
```

recv-buffer-size (IPv6 RIPng)

Use this command to configure the RIPng UDP (User Datagram Protocol) receive-buffer size. This should improve UDP reliability by avoiding UDP receive buffer overruns.

Use the **no** variant of this command to unset the configured RIPng UDP receive-buffer size and set it back to the system default of 196608 bits.

Syntax

```
recv-buffer-size <8192-2147483647>
no recv-buffer-size [<8192-2147483647>]
```

Default The RIPng UDP receive-buffer-size is 196608 bits by default, and is reset to the default using the **no** variant of this command.

Mode Router Configuration

Examples To configure the RIPng UPD, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# recv-buffer-size 23456789

awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# no recv-buffer-size 23456789

awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# no recv-buffer-size
```

redistribute (IPv6 RIPng)

Use this command to redistribute information from other routing protocols into RIPng.

Use the **no** variant of this command to disable the specified redistribution. The parameters **metric** and **routemap** may be used on this command, but have no effect.

Syntax

```
redistribute {connected|static|ospf} [metric <0-16>]
    [route-map <route-map>]
```

```
no redistribute {connected|static|ospf} [metric <0-16>] [route-map
    <route-map>]
```

Parameter	Description
<0-16>	Optional. Specifies the metric value to be used when redistributing information. If a value is not specified, and no value is specified using the **default-metric (IPv6 RIPng)** command, the default is one.
<route-map>	Optional. Specifies route-map to be used to redistribute information.
`connected`	Redistribute from connected routes.
`static`	Redistribute from static routes.
`ospf`	Redistribute from Open Shortest Path First (OSPF).

Default By default, the RIPng metric value is set to 1.

Mode Router Configuration

Example To redistribute information from other routing protocols into RIPng, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# redistribute static route-map mymap
awplus(config-router)# redistribute static metric 8
```

Related Commands **default-metric (IPv6 RIPng)**

route (IPv6 RIPng)

Use this command to configure static RIPng routes.

Use the **no** variant of this command to disable this function.

Syntax

```
route <ipv6-addr/prefix-length>
no route <ipv6-addr/prefix-length>
```

Parameter	Description
<ipv6-addr/prefix-length>	Specify the IPv6 Address in format *X:X::X:X/Prefix-Length*. The prefix-length is a decimal integer between 1 and 128.

Mode Router Configuration

Usage Use this command to add a static RIPng route. After adding the RIPng route, the route can be checked in the RIPng routing table.

Example To configure static RIPng routes, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# route 2001:db8::1/64
```

Related Commands **show ipv6 rip**
clear ipv6 rip route

router ipv6 rip

Use this global command to enter Router Configuration mode to enable a RIPng routing process.

Use the **no** variant of this command to disable the RIPng routing process.

Syntax

```
router ipv6 rip
no router ipv6 rip
```

Mode Global Configuration

Example To enable a RIPng routing process, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)#
```

show debugging ipv6 rip

Use this command to display the RIPng debugging status for the debugging options of: nsm debugging, RIPng event debugging, RIPng packet debugging, and RIPng nsm debugging.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging ipv6 rip`

Mode User Exec and Privileged Exec

Usage Use this command to display the debug status of RIPng.

Example To display the RIPng debugging status, use the following command:

```
awplus# show debugging ipv6 rip
```

show ipv6 protocols rip

Use this command to display RIPng process parameters and statistics.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**

Syntax `show ipv6 protocols rip`

Mode User Exec and Privileged Exec

Example To display RIPng process parameters and statistics, use the following command:

```
awplus# show ipv6 protocols rip
```

Output

```
awplus#show ipv6 protocols rip
Routing Protocol is "RIPng"
  Sending updates every 30 seconds with +/-5 seconds, next due
in 6 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribute metric is 1
  Redistributing:
  Interface
    vlan3
  Routing for Networks:
    fe80::200:cdff:fe27:c086 vlan1
```

show ipv6 rip

Use this command to show RIPng routes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 rip
```

Mode

User Exec and Privileged Exec

Example

To display RIPng routes, use the following command:

```
awplus# show ipv6 rip
```

Output

```
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, Ra - RIP
aggregated, Rcx - RIP connect suppressed, Rsx - RIP static
suppressed, C - Connected, S - Static, O - OSPF

   Network           Next Hop          If     Met Tag   Time

R  2001:db8:1::/48   2001:db8:2::/48   vlan3   3   0    02:28
C  2001:db8:3::/48   ::                vlan2   1   0
Ra 2001:db8:4::/48                     --      1   0
Rs 2001:db8:5::/48   2001:db8:1::/48   vlan3   3   0    02:32
Cs 2001:db8:6::/48   ::                vlan3   1   0
```

Related Commands

show ipv6 rip database

show ipv6 rip database

Use this command to display information about the RIPng database.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 rip database [full]`

Parameter	Description
`full`	Display all IPv6 RIPng full database entries including sub-optimal routes.

Mode User Exec and Privileged Exec

Example To display information about the RIPng database, use the following command:

```
awplus# show ipv6 rip database
```

Output

```
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, Ra - RIP
aggregated, Rcx - RIP connect suppressed, Rsx - RIP static
suppressed, C - Connected, S - Static, O - OSPF

   Network            Next Hop          If      Met Tag   Time

R  2001:db8:1::/48    2001:db8:2::/48   vlan3    3   0    02:28
C  2001:db8:3::/48    ::                vlan2    1   0
Ra 2001:db8:4::/48                      --       1   0
Rs 2001:db8:5::/48    2001:db8:1::/48   vlan3    3   0    02:32
Cs 2001:db8:6::/48    ::                vlan3    1   0
```

Related Commands **show ipv6 rip**

show ipv6 rip interface

Use this command to display information about the RIPng interfaces. You can specify an interface name to display information about a specific interface.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 rip interface [<interface>]`

Parameter	Description
<interface>	The interface to display information about. For instance: `vlan2`.

Mode User Exec and Privileged Exec

Example To display RIPng interface information, use the following command:

```
awplus# show ipv6 rip interface
```

Output

```
lo is up, line protocol is up
RIPng is not enabled on this interface
vlan1 is up, line protocol is up
RIPng is not enabled on this interface
vlan2 is down, line protocol is down
RIPng is not enabled on this interface
vlan3 is up, line protocol is up
Routing Protocol: RIPng
Passive interface: Disabled
Split horizon: Enabled with Poisoned Reversed
IP interface address:
2001:db8:1::1/64
2001:db8:1::2/64
```

timers (IPv6 RIPng)

Use this command to adjust the RIPng routing network timers.

Use the **no** variant of this command to restore the defaults.

Syntax

```
timers basic <update> <timeout> <garbage>
no timers basic
```

Parameter	Description
<update>	<5-2147483647> Specifies the RIPng routing table update timer in seconds. The default is 30 seconds.
<timeout>	<5-2147483647> Specifies the RIPng routing information timeout timer in seconds. The default is 180 seconds. After this interval has elapsed and no updates for a route are received, the route is declared invalid.
<garbage>	<5-2147483647> Specifies the RIPng routing garbage collection timer in seconds. The default is 120 seconds.

Default The default RIPng routing table update timer default is 30 seconds, the default RIPng routing information timeout timer is 180 seconds, and the default RIPng routing garbage collection timer is 120 seconds. The **no** variant of this command restores the default RIPng routing timers.

Mode Router Configuration

Example To adjust the RIPng routing network timers, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 rip
awplus(config-router)# timers basic 30 180 120
```

undebug ipv6 rip

Use this command to disable debugging options of RIPng events, RIPng packets, and communication between RIPng and NSM processes.

Syntax `undebug ipv6 rip [all|events|nsm|packet [recv|send][detail]]`

Parameter	Description
`all`	Disables all RIPng debugging.
`events`	Disable the display of RIPng events information.
`nsm`	Disable the display of RIPng and NSM communication.
`packet`	Disable debugging of specified RIPng packets only.
`recv`	Disable the display of information for received packets.
`send`	Disable the display of information for sent packets.
`detail`	Disable the display of detailed information for sent or received packets.

Mode Privileged Exec and Global Configuration

Example To disable debugging options, use the following command:

```
awplus# undebug ipv6 rip events

awplus# undebug ipv6 rip all

awplus# undebug ipv6 rip packet send

awplus# undebug ipv6 rip packet recv detail
```

Related Commands **debug ipv6 rip**

Chapter 41: OSPF Introduction and Configuration

OSPF Introduction 41.2
Features 41.2
OSPF Components 41.2
- Autonomous Systems 41.2
- Routing Areas 41.3
- Adjacencies and Designated Routers 41.3
- Link State Advertisements 41.4
- OSPF Packet Types 41.4
- OSPF States 41.5
- OSPF Metrics 41.6
- Automatic Cost Calculation 41.7
- Routing with OSPF 41.7
- Network Types 41.8
- Passive Interfaces 41.8
- Authenticating OSPF 41.9
- Redistributing External Routes 41.9

Enabling OSPF on an Interface 41.10
Setting Priority 41.12
Configuring an Area Border Router 41.14
OSPF Cost 41.15
Configuring Virtual Links 41.18
OSPF Authentication 41.20
OSPF Multi-Area Loopback Configuration 41.23

OSPF Introduction

To use this feature, you need a Premium license. Premium licenses will be available from software version 5.4.5 onwards. Please contact your authorized Allied Telesis representative for more information.

This chapter introduces OSPF followed by basic configuration examples. To see details on the OSPF commands used in these examples, or to see the outputs of the validation commands, refer to **Chapter 42, OSPF Commands**.

Features

Open Shortest Path First (OSPF) is an Interior Gateway Routing Protocol, based on Shortest Path First (SPF) or link-state technology. OSPF is defined in RFCs 1245–1247, 1253 and 1583. OSPF was designed specifically for the TCP/IP Internet environment, and supports the following features:

- Authentication of routing updates.
- Tagging of externally-derived routes.
- Fast response to topology changes with low overhead.
- Load sharing over meshed links.

OSPF Components

Autonomous Systems

In SPF-based routing protocols, routers combine to form an Autonomous Systems (AS). These are router systems which operate under a common administration and usually share common routing protocols. Each router maintains a database describing the AS's topology. Each router has an identical database, each component of which describes a particular router and its current state. This includes the state of the interfaces, reachable neighbors, and other information. Information about the AS is distributed between the routers by a process known as "flooding".

Each router runs a routing algorithm, and from the information exchanged about the other AS routers, creates an internal tree-like database of shortest paths with itself as the root. The tree contains a route to each destination in the AS. External routes are added to the tree as "leaves".

Another feature of OSPF is that it enables IP subnets to be configured in a very flexible way. Each route distributed by OSPF has a destination and a mask. During the routing process, routes with the longest mask to a destination are used in preference to those with shorter masks. Host routes are also supported by OSPF; these are considered to be subnets with masks of all ones.

All OSPF protocol exchanges can be authenticated so that only trusted routers participate in the creation of the topology database, and hence the AS's routing. Authentication is disabled by default.

Externally derived routing data can be passed into the AS transparently. The externally derived routing information is kept separate from the OSPF protocol's link state data.

Routing Areas

OSPF allows the grouping of networks into a set, called an **Area**. The internal topology of an area is hidden from the rest of the AS. This technique minimizes the routing traffic required for the protocol. When multiple areas are used, each area has its own copy of the topological database.

Routing can be between areas (inter-area routing) or within areas (intra-area routing). To link together multiple areas, OSPF uses the concept of a **backbone area**. The backbone area forms a central network that links to the other areas within the AS. The backbone must have contiguous connectivity with the its other areas. Virtual links can be used to make the backbone contiguous.

At the junction of each area and its backbone is a **border router**. Packets traveling between areas will do so via the backbone area. Packets are first sent to the area's border router where they will be routed onto the backbone. They will then travel through the backbone to another area border router at the connection to the destination area. The packets are then routed through the destination area to their specific destination.

Adjacencies and Designated Routers

OSPF creates adjacencies between neighboring routers. The reason for forming adjacencies is to exchange topological information. Not every router needs to become adjacent to every other router. Adjacencies are established and maintained with **hello packets**. These packets are sent periodically on all router interfaces. Bidirectional communication is determined by a router seeing itself listed in hello packets from its neighbors. On broadcast multi-access networks, one of the routers becomes a designated router.

The designated router maintains adjacencies with all the routers within the area by issuing link state advertisements for its area and subnet.

The designated router becomes adjacent to all other routers within the area. Since the topological database is spread over adjacencies, the designated router coordinates the synchronization of the topological database on all the routers within its area.

Selecting the designated router

The designated router for a broadcast network is determined dynamically via hello packets. Each router is configured with a priority number, which is advertised in the hello packet. The routers compare their priority numbers, and the router with the highest priority number is elected the designated router. Where routers share the same priority number, as could happen with a common default setting, then the designated router is the router with the highest router ID. On non-broadcast multi-access networks, static configuration information is used to initiate the search for a designated router.

To help in dynamic fallover, OSPF also determines a backup designated router for a network via hello packets. The backup designated router, like the designated router maintains an adjacency to all other routers on the network. If the designated router fails for any reason, the backup designated router takes over.

Link State Advertisements

Link state advertisements are records in the topological database. Routers may generate five different types of link state advertisements **Table 41-1 on page 41.4**. Each type of link state advertisement describes a different set of features of the Autonomous System (AS).

Link state advertisements age to a maximum age called MaxAge (3600 seconds) while stored in the topological database. When a link state advertisement reaches MaxAge, the router tries to flush it from the routing domain by reflooding the advertisement. A link state advertisement that has reached MaxAge is not used in further routing table calculations. The MaxAge link state advertisement is removed totally from the topological database when it is no longer contained on a neighbor link state retransmission list or none of the neighbors are in exchange or loading state. It is relatively rare for a link state advertisement to reach MaxAge because advertisements are usually replaced by more recent instances by normal refresh processes.

Table 41-1: OSPF link state advertisement type

LSA Type	Meaning
Router Links	The router originates a router links advertisement for each area to which it belongs. The advertisement describes the collected states of the router's links to the area. This advertisement also indicates whether the router is an area border router or an AS boundary router.
Network Links	A network link advertisement is originated for every transit multi-access network. This advertisement is originated by the designated router for the transit network, and describes all the OSPF routers fully adjacent to the designated router.
Summary Links	Summary Link advertisements describe a single route to a destination. The destinations described are external to the area, but internal to the AS. Some condensing of routing information occurs when creating these summary link state advertisements.
AS Summary Links	These are like summary link advertisements, but they describe routes to AS boundary routers.
AS External Links	AS external advertisements describe routes external to the AS.

OSPF Packet Types

The OSPF protocol runs directly over IP, using the assigned number 89. The following table describes OSPF packet types.

Table 41-2: OSPF Packet Types

Packet Type	Purpose
Hello	Used to discover and maintain neighbors.
Link State Request	Used to form adjacencies. The router summarizes all its link state advertisements and passes this information, via database description packets, to the router it is forming an adjacency with.
Link State Update	Used for transmission of link state advertisements between routers. This could be in response to a link state request packet or to flood a new or more recent link state advertisement.
Link State Acknowledgement	Used to make the flooding of link state advertisements reliable. Each link state advertisement received is explicitly acknowledged.

OSPF States

The following table describes the states that neighbors can be in:

Table 41-3: OSPF States

Packet Type	Purpose
Down	This is the initial state. No hello packets have been received from the neighbor recently or at all.
Attempt	This state applies to non-broadcast multi-access networks. The router is making a determined attempt to contact a statically configured neighbor. Hello packets are sent every hello interval.
Init	A hello packet has been seen from the neighbor. However, the hello packet does not list the router as known.
2-Way	This state is entered when the communication between to neighbors is bidirectional (the hello packet from the neighbor lists this router as a neighbor).
ExStart	This is the first step in creating an adjacency between two routers. The two routers decide which is going to control the exchange between them.
Exchange	In this state, the neighbors exchange database description packets. Each packet summarizes the link state advertisements held by that router.
Loading	After all the database description information has been exchanged, the routers exchange link state advertisements required to update or complete each router's topological database thereby synchronizing the two router's databases.
Full	This is the final state and the adjacency is complete. Reaching this state in itself may cause new instances of some link state advertisements, such as the network and router advertisements related to the two routers.

The following table describes the interface states that the router can be in.

Table 41-4: OSPF Interface States

Packet Type	Purpose
Down	The initial state. No traffic can be routed with the interface in this state.
Loopback	The router's interface to the network is looped back.
Waiting	Interfaces to broadcast and non-broadcast multi-access networks enter this state when they are started. In this state the router tries to determine the designated or backup designated router. The router is not allowed to elect a backup or designated router while in the waiting state. This stops unnecessary changes in the designated router.

Table 41-4: OSPF Interface States(cont.)

Packet Type	Purpose
Point-to-Point	The interface is operational and is connected to a point-to-point network.
DROther	The interface to a broadcast or a non-broadcast multi-access network has not been selected as either the designated router or backup designated router for the network.
Backup	The interface to a broadcast or a non-broadcast multi-access network has been selected as the backup designated router for the network.
DR	The interface to a broadcast or a non-broadcast multi-access network has been selected as the designated router for the network.

OSPF Metrics

The metrics used by OSPF are not simple distance metrics, such as used by RIP for example, but are measurements of the path bandwidth. Interface metrics are normally set using the formula 10^8 / interface speed (in bps). This gives metrics such as 1 for a 100 Mbps Ethernet interface, and 1562 for a 64 kbps serial line.

When no metric is defined, the default metric for routes redistributed into OSPF is 20. You can define the metric using the **redistribute (OSPF)** command.

You can also define the metric using the **set metric** command for a route map. If a route map is configured, but no OSPF metric is defined using a **redistribute (OSPF)** command, then a default metric of 20 is used. If a route map is configured and the metric is defined using the **set metric** command, this supersedes a metric defined using the **redistribute (OSPF)** command.

How to set the OSPF metric when redistributing routes

You can determine the OSPF metric using the following methods:

- use the default OSPF metric, which is 20.
- change it using the **metric** parameter of the **redistribute (OSPF)** command

The following example sets the metric to `5` for `connected` redistributed routes:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# redistribute connected metric 5
```

- change it using the **metric-value** parameter of the **set metric** command for a route map

Note that changing the metric using a route map supersedes the metric defined using the **redistribute (OSPF)** command. The following example for entry `3` of the route map called `rmap1` give matching routes a metric of `5`:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# set metric 5
```

Automatic Cost Calculation

OSPF interfaces can automatically set the OSPF metric of an IP interface based on its bandwidth, instead of the system administrator having to manually set the OSPF metric. Automatic setting takes into account that the speed of an interface can change over time, when ports change link state or change speed via auto-negotiation or manual setting. If metrics are manually set, some interfaces are preferred when they should be changing to match dynamically changing network configurations.

Routing with OSPF

To route an IP packet, the router looks up the routing table for the entry that best matches the destination of the packet. This entry contains the interface and next hop router required to forward the IP packet to its destination. The entry that best matches the destination is determined first by the path type, then the longest (most specific) network mask. At this point there may still be multiple routing entries to the destination; if so, then equi-cost multi-path routes exist to the destination. Table **Table 41-5 on page 41.7** shows the available path types used when routing packets.

Table 41-5: OSPF Path Types

Path Type	Description
Intra	Route to the destination is within a single OSPF area.
Inter	Route to the destination is within the AS, but spans more than one OSPF area.
Ext1	Route to the destination is via an AS router within the AS. This is an OSPF external route of Type 1. Type 1 external routes add the external metric (as received by the AS router), and the internal OSPF metric to reach the AS router, to determine the final metric to the destination.
Ext2	Route to the destination is via an AS router within the AS. This is an OSPF external route of Type 2. Type 2 external routes use only the external path cost to determine the preferred route. Internal metrics are only used where two or more interfaces present the same external path cost.

Network Types

OSPF treats the networks attached to OSPF interfaces as one of the following network types, depending on the physical media:

- broadcast
- non-broadcast multi-access (nbma)
- point-to-point
- point-to-multipoint
- virtual

By default, VLAN and Ethernet networks are treated as broadcast networks. You can use the **ip ospf network** command to configure a VLAN interface to be other network types. Configure a VLAN or Ethernet interface as an NBMA interface when:

- Some devices on the network do not support multicast addressing.
- You want to select which devices on the network are to become OSPF neighbors, rather than allow all the devices on the network to become OSPF neighbors.

Passive Interfaces

A passive interface does not take part in normal OSPF interface operations:

- OSPF does not transmit or receive Hello messages via the interface.
- The interface does not experience interface state transitions.
- OSPF does not associate neighbors with the interface.

If the interface is up, OSPF adds the network attached to the interface as a stub network to the router LSA of the area in which the interface resides.

Usage Configure an interface to be passive if you wish its connected route to be treated as an OSPF route (rather than an AS-external route), but do not wish to actually exchange any OSPF packets via this interface.

Examples To configure passive interface mode on interface **vlan2**, enter the following commands:

```
awplus(config)# router ospf 100
awplus(config-router)# passive-interface vlan2
```

To configure passive interface mode on **all** interfaces, enter the following commands:

```
awplus(config)# router ospf 100
awplus(config-router)# passive-interface
```

Authenticating OSPF

OSPF packet authentication is described in Appendix D of RFC 2328, and in RFC 1583. RFC 2328 describes authentication set for an interface, whilst RFC 1583 describes authentication set per area. Refer to the RFCs for a detailed description of these methods of authentication.

There are two ways to authenticate an OSPF packet:

- password authentication
- cryptographic authentication

Use the following commands to configure authentication on a specific VLAN interface:

- **ip ospf authentication**
- **ip ospf authentication-key**
- **ip ospf message-digest-key**

Use the following commands to configure authentication for a specific OSPF area.

- **area authentication**
- **ip ospf message-digest-key**

Redistributing External Routes

OSPF can import and redistribute RIP, non-OSPF interface, and statically configured routes. It can also optionally assign any of the following settings to all routes it imports:

- a route metric
- the External metric type
- a tag—a number to label the route

Alternatively, you can assign a route map to select particular routes and set their route parameters. A route map can also filter out a subset of routes, so you do not have to import all routes.

The import settings also allow you to select whether to redistribute subnets (classless network routes), or only classful network routes.

To import and redistribute external routes into OSPF, create a route redistribution definition for the source routing protocol, using the **redistribute (OSPF)** command.

Summarizing routes for redistribution

OSPF can summaries external routes to be redistributed using a list of administratively defined summary addresses specified as network/mask pairs. The summary addresses replace the original routes in AS external LSAs. Use summary addresses to reduce the number of AS external routes advertised by the router.

You can set the following attributes for summary addresses:

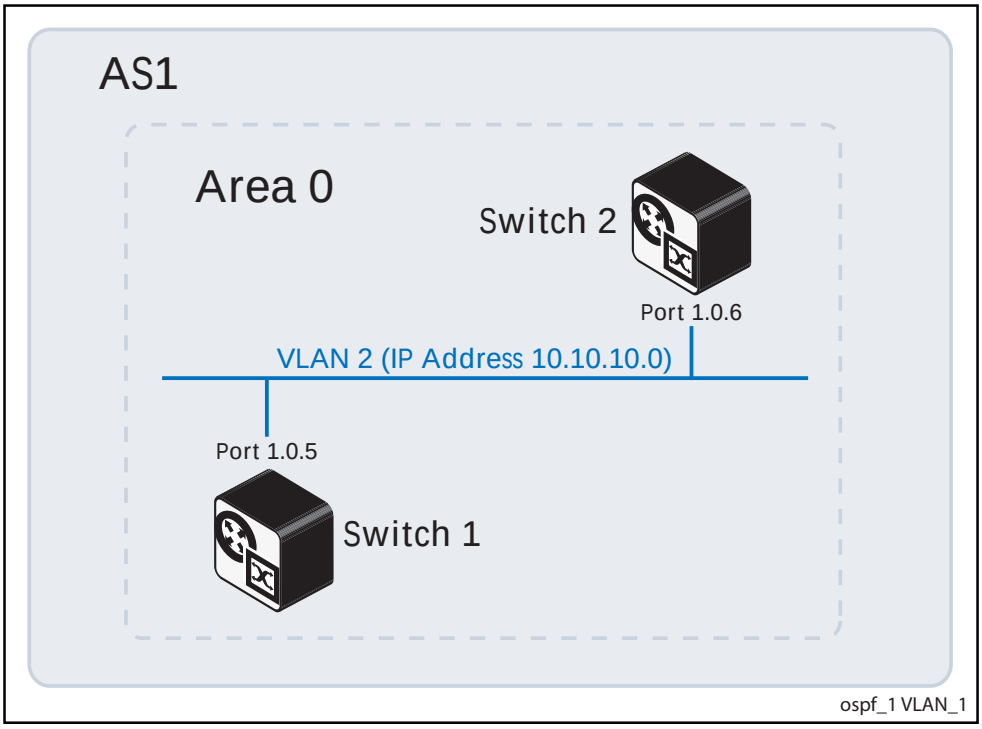
- Whether the summary address is advertised.
- The tag to be inserted in the AS external LSA. The tag overrides tags set by the original route used to select the original routes for redistribution.

To create summary addresses for route redistribution, use the **summary-address** command.

Enabling OSPF on an Interface

This example shows the minimum configuration required for enabling OSPF on an interface. In this example, the OSPF routers are Allied Telesis managed Layer 3 switches. `Switch 1` and `Switch 2` are two OSPF routers in `Area 0` connecting to network 10.10.10.0/24.

Note Configure one interface so that it belongs to only one area. However, you can configure different interfaces on an OSPF router to belong to different areas.



Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `interface port1.0.5`	Set the switchport mode to **access**
`awplus#` `switchport access vlan2`	Assign port 1.0.5 to VLAN 2
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`)
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define the interface (`10.10.10.0/24`) on which OSPF runs and associate the area ID (`0`) with the interface (area ID `0` specifies the backbone area).

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode
`awplus(config)#` `interface port1.0.6`	Set the switchport mode to **access**
`awplus#` `switchport access vlan2`	Assign port 1.0.6 to VLAN 2
`awplus(config)#` `router ospf 200`	Configure the Routing process and specify the Process ID `(200)`. The Process ID should be a unique positive integer identifying the routing process. Note that the process ID used on this switch is different to that used on Switch 1. This is correct configuration as the process ID is a value that is only used within a single OSPF router. Therefore there is no requirement for the process IDs used on one OSPF router to have any relationship with the process IDs used on the other OSPF routers that it interacts with.
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define the interface (`10.10.10.0/24`) on which OSPF runs and associate the area ID (`0`) with the interface.

Names of commands used

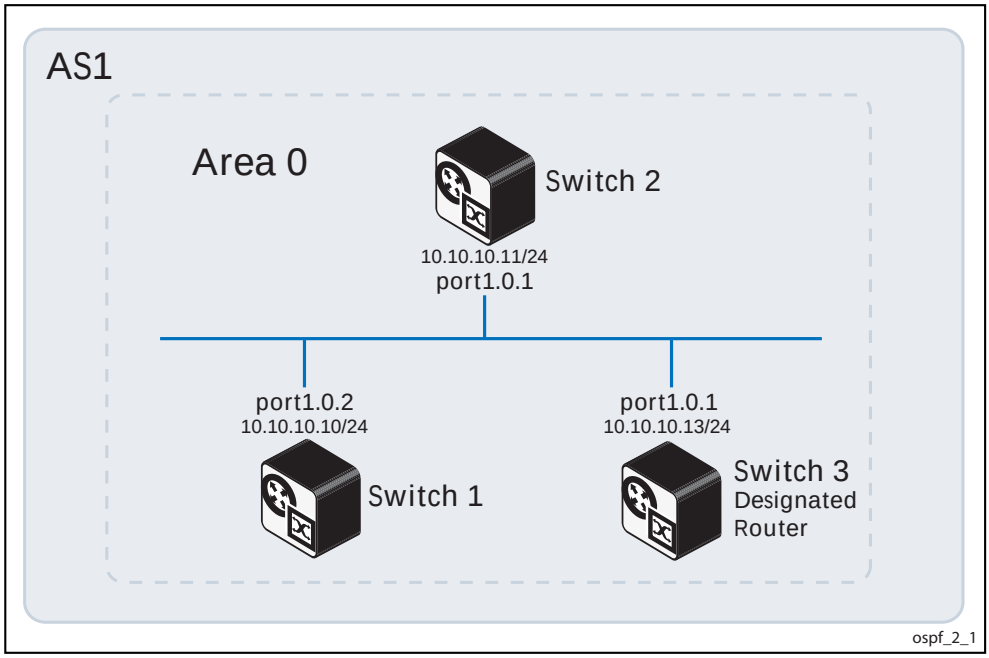
network area
router ospf

Validation commands

show ip ospf
show ip ospf interface
show ip ospf neighbor
show ip ospf route

Setting Priority

This example shows the configuration for setting the priority for an interface. You can set a high priority for an OSPF router to make it the Designated Router (DR). In this example, the OSPF routers are Allied Telesis managed Layer 3 switches. Switch 3 is configured to have a priority of 10, which is higher than the default priority (default priority is 1) of Switch 1 and Switch 2; making it the DR. In this example network the back-up DR would be Switch 2 as it has a higher router ID than Switch 1.



Switch 3

`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) to be configured.
`awplus(config-if)#` `ip ospf priority 10`	Specify the router priority to a higher priority (`10`) to make Switch 3 the Designated Router (DR).
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define the interface (`10.10.10.0/24`) on which OSPF runs and associate the area ID (`0`) with the interface.

Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define the interface (`10.10.10.0/24`) on which OSPF runs and associate the area ID (`0`) with the interface (area ID 0 specifies the backbone area).

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 200`	Configure the Routing process and specify the Process ID (200). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define the interface (`10.10.10.0/24`) on which OSPF runs and associate the area ID (`0`) with the interface.

Names of commands used

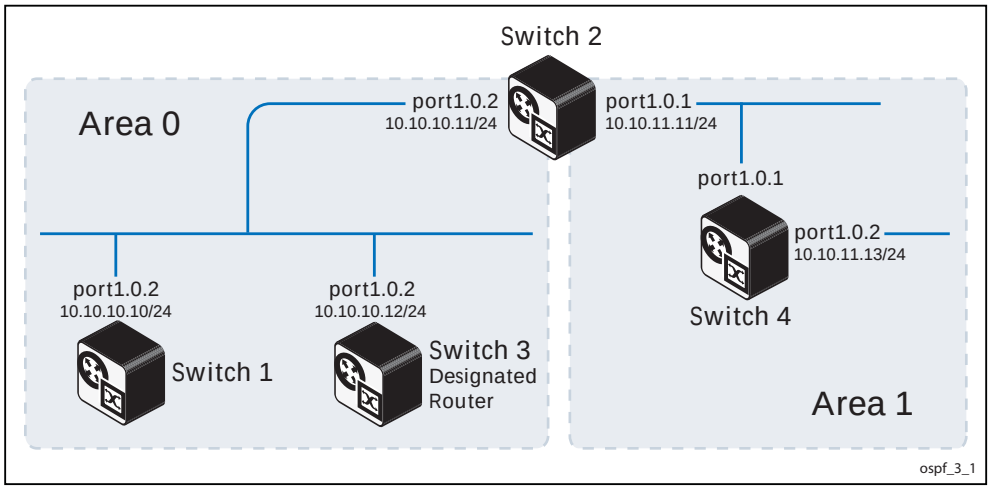
network area
ip ospf priority

Validation commands

show ip ospf neighbor
show ip ospf interface

Configuring an Area Border Router

This example shows configuration for an Area Border Router (ABR). In this example, the OSPF routers are Allied Telesis managed Layer 3 switches. `Switch 2` is an ABR, where interface `vlan2` is in `Area 0` and interface `vlan3` is in `Area 1`.



Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define one interface (`10.10.10.0/24`) on which OSPF runs and associate the area ID (`0`) with the interface.
`awplus(config-router)#` `network 10.10.11.0/24 area 1`	Define the other interface `(10.10.11.0/24)` on which OSPF runs and associate the area ID `(1)` with the interface.

Names of commands used

network area

Validation commands

show ip ospf
show ip ospf interface

OSPF Cost

You can make a route the preferred route by changing its cost. In this example, the OSPF routers are Allied Telesis managed Layer 3 switches. The cost has been configured to make `Switch 2` the next hop for `Switch 1`.

The default cost on each interface is 10. Interface `vlan2` on `Switch 2` has a cost of 100 and interface `vlan3` on Switch 3 has a cost of 150. The total cost for Switch 1 to reach 10.10.14.0/24 (Switch 4) `Switch 2` or via `Switch 3` is:

Switch 2: 10+100 = 110

Switch 3: 10+150 = 160

Therefore, `Switch 1` chooses `Switch 2` as its next hop for destination 10.10.14.0/24, as that path has the lower cost.

Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.

`awplus(config-router)#`
`network 10.10.9.0/24 area 0`
`awplus(config-router)#`
`network 10.10.10.0/24 area 0`
`awplus(config-router)#`
`network 10.10.12.0/24 area 0`
	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface (area ID 0 specifies the backbone area).

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) to be configured.

`awplus(config-if)#` `ip ospf cost 100`	Set the OSPF cost of this link to `100`.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (100). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `network 10.10.10.0/24 area 0`	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface.
`awplus(config-router)#` `network 10.10.11.0/24 area 0`	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface.

Switch 3

`awplus(config)#` `interface vlan3`	Specify the interface (`vlan3`) to be configured.
`awplus(config-if)#` `ip ospf cost 150`	Set the OSPF cost of this link to `100`.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `network 10.10.12.0/24 area 0`	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface.
`awplus(config-router)#` `network 10.10.13.0/24 area 0`	

Switch 4

`awplus(config)#`	
`router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#`	
`network 10.10.11.0/24 area 0`	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface.
`awplus(config-router)#`	
`network 10.10.13.0/24 area 0`	
`awplus(config-router)#`	
`network 10.10.14.0/24 area 0`	

Names of commands used

network area
ip ospf cost

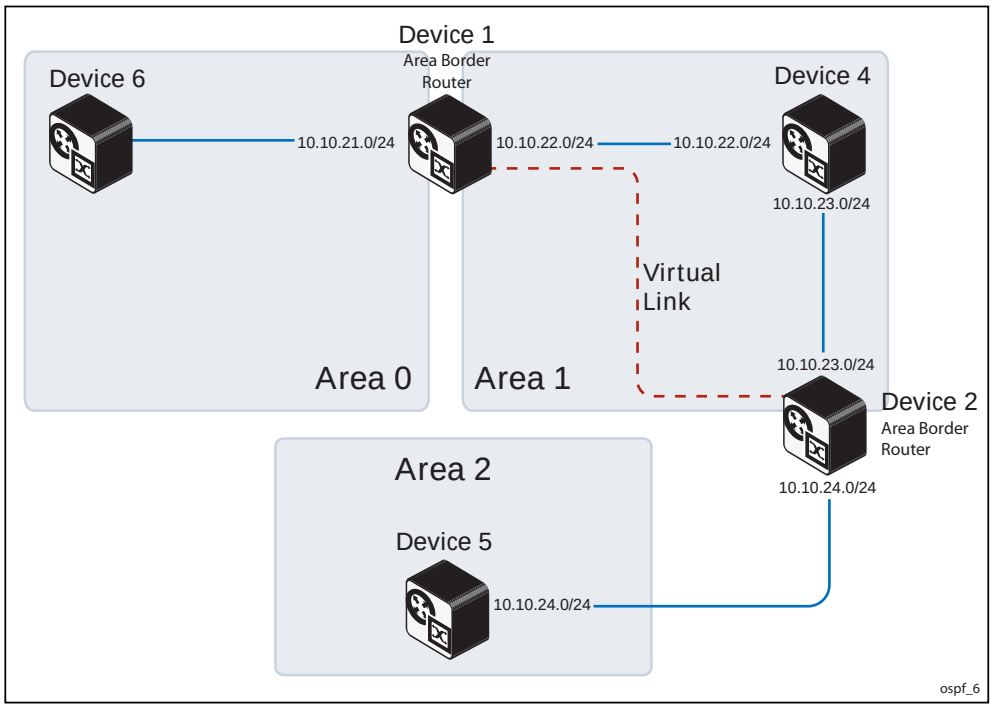
Validation commands

show ip ospf route

Configuring Virtual Links

Virtual links are used to connect a disjointed non-backbone area to the backbone area, or to repair a non-contiguous backbone area. In this example, the OSPF routers shown represent any Allied Telesis managed Layer 3 switches or Allied Telesis routers.

In the network below, there is no area border router that connects Area2 to the backbone. So a virtual link needs to be created between ABR Device 1 and ABR Device 2 to connect `Area 2` to `Area 0`. Area 1 is used as a transit area.



Device 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `ospf router-id 10.10.21.1`	Configure OSPF Router ID (`10.10.21.1`) for this router.

`awplus(config-router)#`
`network 10.10.21.0/24 area 0`
`awplus(config-router)#`
`network 10.10.22.0/24 area 1`
	Define interfaces on which OSPF runs and associate the area IDs (`0` and `1`) with the interface.

`awplus(config-router)#` `area 1 virtual-link 10.10.23.2`	Configure a virtual link between Device 1 and Device 2 (`10.10.23.2`) through transit area 1.

Device 2

`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (`100`). The Process ID should be a unique positive integer identifying the routing process.
`awplus(config-router)#` `ospf router-id 10.10.23.2`	Configure OSPF Router ID (`10.10.23.2`) for this router.

`awplus(config-router)#`
`network 10.10.23.0/24 area 1`
`awplus(config-router)#`
`network 10.10.24.0/24 area 2`
	Define interfaces on which OSPF runs and associate the area IDs (`1` and `2`) with the interface.

`awplus(config-router)#` `area 1 virtual-link 10.10.21.1`	Configure a virtual link between Device 2 and Device 1 (`10.10.21.1`) through transit area 1.

Names of commands used

area virtual-link
network area

Validation commands

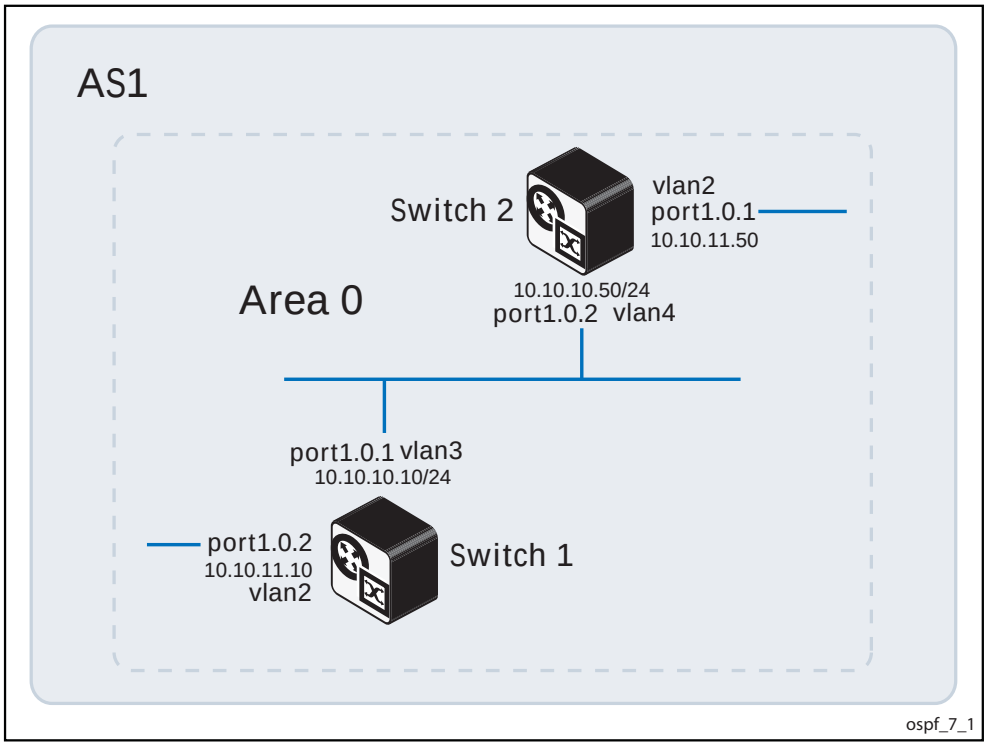
show ip ospf virtual-links
show ip ospf neighbor
show ip ospf
show ip ospf route

OSPF Authentication

In the AlliedWare Plus™ implementation there are three types of OSPF authentications--Null authentication (Type 0), Simple Text (Type 1) authentication and MD5 (Type 2) authentication. With null authentication, routing exchanges over the network are not authenticated. In Simple Text authentication, the authentication type is the same for all OSPF routers that communicate using OSPF in a network. For MD5 authentication, you configure a key and a key-id on each OSPF router. The OSPF router generates a message digest on the basis of the key, key ID and the OSPF packet and adds it to the OSPF packet.

The Authentication type can be configured on a per-interface basis or a per-area basis. Additionally, Interface and Area authentication can be used together. Area authentication is used for an area and interface authentication is used for a specific interface in the area. If the Interface authentication type is different from Area authentication type, Interface authentication type overrides the Area authentication type. If the Authentication type is not specified for an interface, the Authentication type for the area is used. The authentication command descriptions contain details of each type of authentication. Refer to **Chapter 42, OSPF Commands** for OSPF authentication commands.

In this example, the OSPF routers are Allied Telesis managed Layer 3 switches. Switch 1 and Switch 2 are configured for both the interface and area authentications. The authentication type of interface `vlan2` on Switch 1 and interface `vlan2` on Switch 2 is md5 mode and is defined by the **area authentication** command on page 42.3; however, the authentication type of interface vlan3 on Switch 1 and interface vlan4 on Switch 2 is plain text mode and is defined by the **ip ospf authentication** command on page 42.34. This interface command overrides the area authentication command.



Switch 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (100). The Process ID should be a unique positive integer identifying the routing process.

`awplus(config-router)#`
`network 10.10.10.0/24 area 0`
`awplus(config-router)#`
`network 10.10.11.0/24 area 0`
	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface (area ID 0 specifies the backbone area).

`awplus(config-router)#` `area 0 authentication message-digest`	Enable MD5 authentication on area 0.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) you are configuring.
`awplus(config-if)#` `ip ospf message-digest-key 1 md5 test`	Register MD5 key `test` for OSPF authentication. The Key ID is 1.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to Global Configuration mode
`awplus(config)#` `interface vlan3`	Specify the interface (`vlan3`) you are configuring.
`awplus(config-if)#` `ip ospf authentication`	Enable OSPF packet to use text authentication on the current interface (`vlan3`).
`awplus(config-if)#` `ip ospf authentication-key test`	Specify an OSPF authentication password test for the neighboring OSPF routers.

Switch 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ospf 100`	Configure the Routing process and specify the Process ID (100). The Process ID should be a unique positive integer identifying the routing process.

`awplus(config-router)#`
`network 10.10.10.0/24 area 0`
`awplus(config-router)#`
`network 10.10.11.0/24 area 0`
	Define interfaces on which OSPF runs and associate the area ID `(0)` with the interface (area ID 0 specifies the backbone area).

`awplus(config-router)#` `area 0 authentication message-digest`	Enable MD5 authentication on area 0.
`awplus(config-router)#` `exit`	Exit the Router Configuration mode and return to the Global Configuration mode.
`awplus(config)#` `interface vlan2`	Specify the interface (`vlan2`) you are configuring.
`awplus(config-if)#` `ip ospf message-digest-key 1 md5 test`	Register MD5 key `test` for OSPF authentication. The Key ID is 1.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and return to Global Configuration mode
`awplus(config)#` `interface vlan4`	Specify the interface (`vlan4`) you are configuring.
`awplus(config-if)#` `ip ospf authentication`	Enable OSPF packet to use text authentication on the current interface (`vlan4`).
`awplus(config-if)#` `ip ospf authentication-key test`	Specify an OSPF authentication password `test` for the neighboring OSPF routers.

Names of commands used

ip ospf authentication
ip ospf authentication-key
network area
area authentication

Validation commands

show running-config
show ip ospf neighbor

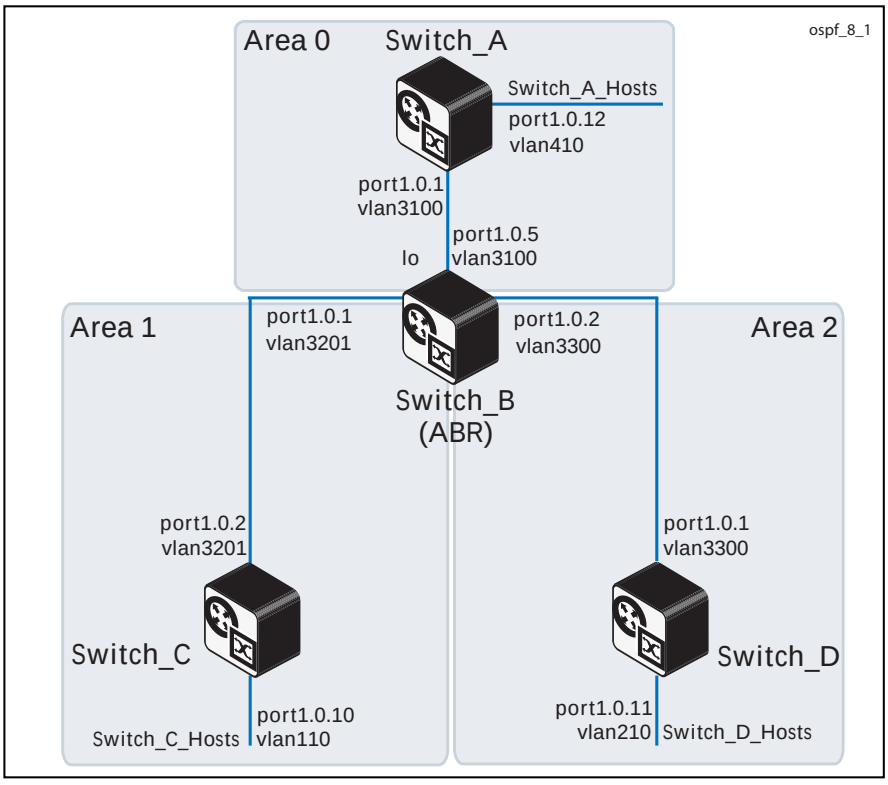
OSPF Multi-Area Loopback Configuration

When a switch is connected to more than two areas, and area 0 interfaces go down, then all routing between the remaining areas will stop. This is due to how OSPF processes inter-area routes as they pass through area 0, even if they are destined for another area. This is a globally recognized issue. Configuring a loopback interface on area 0 resolves this issue.

This example OSPF multiple area configuration applies the best practice of configuring a loopback interface in area 0. This is done so that the ABR (Area Border Router), which in this example is Switch_B, remains available to help connected areas exchange traffic. If the physical link fails between Switch_A and Switch_B, area 0 remains available on Switch_B.

An ABR is a member of both the OSPF backbone and the attached areas. ABRs maintain routing tables describing both the backbone topology and the topology of other areas.

LDS (Long Distance Stacking) enables networks where multiple switches may connect to multiple areas with more than two areas configured on an ABR. Configuring a loopback on the ABR prevents all the inter-area and intra-area OSPF area routing from stopping if the interfaces in an area go down. The area stays up when the interfaces of the area go down.



Switch_A configuration

See the below OSPF configuration for a device with the hostname **Switch_A**:

```
hostname Switch_A
!
vlan database
 vlan 410 name Switch_A_Hosts
 vlan 3100 name Area0
!
interface port1.0.1
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 3100
 switchport trunk native vlan none
!
interface port1.0.12
 switchport
 switchport mode access
 switchport access vlan 410
!
interface vlan410
 ip address 172.16.131.65/29
!
interface vlan3100
 description << Area 0 >>
!
router ospf 100
 ospf router-id 0.0.0.2
 passive-interface vlan410
 network 172.16.128.16/29 area 0
 network 172.16.131.64/29 area 0
!
```

Switch_B configuration

See the below OSPF configuration for a device with the hostname **Switch_B**:

```
hostname Switch_B
!
vlan database
 vlan 3100 name Area0
 vlan 3201 name Area1
 vlan 3300 name Area2
!
interface port1.0.1
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 3201
 switchport trunk native vlan none
!
interface port1.0.5
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 3100
 switchport trunk native vlan none
!
interface port1.0.2
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 3300
 switchport trunk native vlan none
!
interface lo
 ip address 1.1.1.1/32
!
interface vlan3100
 description << Area 0 >>
 ip address 172.16.128.17/29
!
interface vlan3201
 description << Area 1 >>
 ip address 172.16.128.9/29
!
interface vlan3300
 description << Area 2 >>
 ip address 172.16.128.1/29
!
router ospf 100
 ospf router-id 0.0.0.1
 passive-interface lo
 network 1.1.1.1/32 area 0
 network 172.16.128.0/29 area 2
 network 172.16.128.8/29 area 1
 network 172.16.128.16/29 area 0
!
```

Switch_C configuration

See the below OSPF configuration for a device with the hostname **Switch_C**:

```
hostname Switch_C
!
vlan database
 vlan 110 name Switch_C_Hosts
 vlan 3201 name Area1
!
interface port1.0.2
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 3201
 switchport trunk native vlan none
!
interface port1.0.10
 switchport
 switchport mode access
 switchport access vlan 110
 spanning-tree edgeport
!
interface vlan110
 description << Switch C Hosts >>
 ip address 172.16.129.1/26
!
interface vlan3201
 description << Area 1 >>
 ip address 172.16.128.10/29
!
router ospf 101
 ospf router-id 1.1.1.1
 passive-interface vlan110
 network 172.16.128.8/29 area 1
 network 172.16.129.0/26 area 1
!
```

Switch_D configuration

See the below OSPF configuration for a device with the hostname **Switch_D**:

```
hostname Switch_D
!
vlan database
  vlan 210 name Switch_D_Hosts
  vlan 3300 name Area2
!
interface port1.0.1
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 3300
 switchport trunk native vlan none
!
interface port1.0.11
 switchport
 switchport mode access
 switchport access vlan 210
!
interface vlan210
 description << Switch D Hosts >>
 ip address 172.16.130.1/25
!
interface vlan3300
 description << Area 2 >>
 ip address 172.16.128.3/29
!
router ospf 102
 ospf router-id 2.2.2.2
 passive-interface vlan210
 passive-interface vlan220
 network 172.16.128.0/29 area 2
 network 172.16.130.0/25 area 2
 network 172.16.130.128/27 area 2
!
```

Chapter 42: OSPF Commands

Introduction 42.3
Command List 42.3
area authentication 42.3
area default-cost 42.5
area filter-list 42.6
area nssa 42.7
area range 42.9
area stub 42.11
area virtual-link 42.12
auto-cost reference bandwidth 42.14
bandwidth 42.16
capability opaque 42.17
capability restart 42.17
clear ip ospf process 42.18
compatible rfc1583 ..... 42.18
debug ospf events 42.19
debug ospf ifsm 42.20
debug ospf lsa 42.21
debug ospf nfsm 42.22
debug ospf nsm 42.23
debug ospf packet 42.24
debug ospf route 42.25
default-information originate (OSPF) 42.26
default-metric (OSPF) 42.27
distance (OSPF) 42.28
distribute-list (OSPF) 42.30
enable db-summary-opt 42.32
host area 42.33
ip ospf authentication 42.34
ip ospf authentication-key 42.35
ip ospf cost 42.36
ip ospf database-filter 42.37
ip ospf dead-interval 42.38
ip ospf disable all 42.39
ip ospf hello-interval 42.40
ip ospf message-digest-key 42.41
ip ospf mtu 42.42
ip ospf mtu-ignore 42.43
ip ospf network 42.44
ip ospf priority 42.45
ip ospf resync-timeout 42.46
ip ospf retransmit-interval 42.47
ip ospf transmit-delay 42.48
max-concurrent-dd 42.49
maximum-area 42.50
neighbor (OSPF) 42.51
network area 42.52

ospf abr-type 42.53
ospf restart grace-period 42.54
ospf restart helper 42.55
ospf router-id 42.57
overflow database 42.58
overflow database external 42.59
passive-interface (OSPF) 42.60
redistribute (OSPF) 42.61
restart ospf graceful 42.63
router ospf 42.64
router-id 42.65
show debugging ospf 42.66
show ip ospf 42.67
show ip ospf border-routers 42.69
show ip ospf database 42.70
show ip ospf database asbr-summary 42.72
show ip ospf database external 42.73
show ip ospf database network 42.75
show ip ospf database nssa-external 42.77
show ip ospf database opaque-area 42.79
show ip ospf database opaque-as 42.80
show ip ospf database opaque-link 42.81
show ip ospf database router 42.82
show ip ospf database summary 42.83
show ip ospf interface 42.85
show ip ospf neighbor 42.86
show ip ospf route 42.88
show ip ospf virtual-links 42.89
show ip protocols ospf 42.90
summary-address 42.91
timers spf exp 42.92
undebug ospf events 42.93
undebug ospf ifsm 42.93
undebug ospf lsa 42.93
undebug ospf nfsm 42.93
undebug ospf nsm 42.93
undebug ospf packet 42.93
undebug ospf route 42.93

Introduction

This chapter provides an alphabetical reference of commands used to configure OSPF. For more information, see **Chapter 41, OSPF Introduction and Configuration**.

Command List

area authentication

Use this command to enable authentication for an OSPF area. Specifying the area authentication sets the authentication to Type 1 authentication or the Simple Text password authentication (details in RFC 2328).

The **no** variant of this command removes the authentication specification for an area.

Syntax

```
area <area-id> authentication [message-digest]
no area <area-id> authentication
```

Parameter	Description	
<area-id>	The OSPF area that you are enabling authentication for. This can be entered in either dotted decimal format or normal decimal format.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address, entered in the form A.B.C.D.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area OSPF Area ID.	
`message-digest`	Enables MD5 authentication in the OSPF area.	

Default By default, no authentication occurs.

Mode Router Configuration

Usage All OSPF packets transmitted in this **area** must have the same password in their OSPF header. This ensures that only routers that have the correct password may join the routing domain.

Give all routers that are to communicate with each other through OSPF the same authentication password.

Use the **ip ospf authentication-key** command to specify a Simple Text password. Use the **ip ospf message-digest-key** command to specify MD5 password.

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# area 1 authentication
```

Related Commands

ip ospf authentication
ip ospf message-digest-key

area default-cost

This command specifies a cost for the default summary route sent into a stub or NSSA area.

The **no** variant of this command removes the assigned default-route cost.

Syntax

```
area <area-id> default-cost <0-16777215>
no area <area-id> default-cost
```

Parameter	Description	
<area-id>	The OSPF area that you are specifying the default summary route cost for. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
`default-cost`	Indicates the cost for the default summary route used for a stub or NSSA area. Default: **1**	

Mode Router Configuration

Usage The default-cost option provides the metric for the summary default route, generated by the area border router, into the NSSA or stub area. Use this option only on an area border router that is attached to the NSSA or stub area. Refer to the RFC 3101 for information on NSSA.

Example To set the default cost to 10 in area 1 for the OSPF instance 100, use the commands:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# area 1 default-cost 10
```

Related Commands **area nssa**
area stub

area filter-list

This command configures filters to advertise summary routes on Area Border Routers (ABR).

This command is used to suppress particular intra-area routes from/to an area to/from the other areas. You can use this command in conjunction with either the access-list or prefix-list command.

The **no** variant of this command removes the filter configuration.

Syntax

```
area <area-id> filter-list {access <access-list>|prefix <prefix-
    list>} {in|out}

no area <area-id> filter-list {access <access-list>|prefix <prefix-
    list>} {in|out}
```

Parameter	Description	
<area-id>	The OSPF area that you are configuring the filter for. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address format `A.B.C.D`.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
`access`	Use access-list to filter summary.	
`prefix`	Use prefix-list to filter summary.	
<access-list>	Name of an access-list.	
<prefix-list>	Name of a prefix-list.	
`in`	Filter routes from the other areas to this area.	
`out`	Filter routes from this area to the other areas.	

Mode Router Configuration

Example To configure filters to advertise summary routes, use the following commands:

```
awplus# configure terminal
awplus(config)# access-list 1 deny 172.22.0.0
awplus(config)# router ospf 100
awplus(config-router)# area 1 filter-list access 1 in
```

area nssa

This command sets an area as a Not-So-Stubby-Area (NSSA). By default, no NSSA area is defined.

Use this command to simplify administration if you are connecting a central site using OSPF to a remote site that is using a different routing protocol. You can extend OSPF to cover the remote connection by defining the area between the central router and the remote router as an NSSA.

There are no external routes in an OSPF stub area, so you cannot redistribute from another protocol into a stub area. A NSSA allows external routes to be flooded within the area. These routes are then leaked into other areas. Although, the external routes from other areas still do not enter the NSSA. You can either configure an area to be a stub area or an NSSA, not both.

The **no** variant of this command removes this designation.

Syntax

```
area <area-id> nssa [default-information-originate <metric> |
   no-redistribution | no-summary | translator-role <role> ]

no area <area-id> nssa [default-information-originate |
   no-redistribution | no-summary | translator-role ]
```

Parameter	Description	
<area-id>	The OSPF area that you are configuring as an NSSA. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address format `A.B.C.D`.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
`default-information-originate`	Originate Type-7 default LSA into NSSA.	
<metric>	The external or internal metric. Specify the following:	
	`metric` *<0-16777214>*	The metric value.
	`metric-type` *<1-2>*	External metric type.
`no-redistribution`	Do not redistribute external route into NSSA.	
`no-summary`	Do not inject inter-area route into NSSA.	
`translator-role`	Specify NSSA-ABR translator-role.	

Parameter(cont.)	Description(cont.)	
<role>	The role type. Specify one of the following keywords:	
	`always`	Router always translate NSSA-LSA to Type-5 LSA.
	`candidate`	Router may translate NSSA-LSA to Type-5 LSA if it is elected.
	`never`	Router never translate NSSA-LSA.

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# area 0.0.0.51 nssa
awplus(config-router)# area 3 nssa translator-role candidate
no-redistribution default-information-
originate metric 34 metric-type 2
```

Related Commands **area default-cost**

area range

Use this command to summarize OSPF routes at an area boundary, configuring an IPv4 address range which consolidates OSPF routes. By default, this feature is not enabled.

A summary route created by this command is then advertised to other areas by the Area Border Routers (ABRs). In this way, routing information is condensed at area boundaries and outside the area so that routes are exchanged between areas in an efficient manner.

If the network numbers in an area are arranged into sets of contiguous routes, the ABRs can be configured to advertise a summary route that covers all the individual networks within the area that fall into the specified range.

The **no** variant of this command disables this function and restores default behavior.

Syntax

```
area <area-id> range <ip-addr/prefix-length> [advertise|not-
    advertise]

no area <area-id> range <ip-addr/prefix-length>
```

Parameter	Description	
<area-id>	The OSPF area that you summarizing the routes for. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address format `A.B.C.D`.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
<ip-addr/prefix-length>	The area range prefix and length.	
`advertise`	Advertise this range as a summary route into other areas.	
`not-advertise`	Does not advertise this range.	

Default The area range is not configured by default. The area range is advertised if it is configured.

Mode Router Configuration

Usage You can configure multiple ranges on a single area with multiple instances of this command, so OSPF summarizes addresses for different sets of IPv4 address ranges.

Ensure OSPF IPv4 routes exist in the area range for advertisement before using this command.

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# area 1 range 192.16.0.0/16
awplus(config-router)# area 1 range 203.18.0.0/16
```

area stub

This command defines an OSPF area as a stub area. By default, no stub area is defined.

Use this command when routers in the area do not require learning about summary LSAs from other areas. You can define the area as a totally stubby area by configuring the Area Border Router of that area using the **area stub no-summary** command.

There are two stub area router configuration commands: the **area stub** and **area default-cost** commands. In all routers attached to the stub area, configure the area by using the **area stub** command. For an area border router (ABR) attached to the stub area, also use the **area default-cost** command.

The **no** variant of this command removes this definition.

Syntax

```
area <area-id>  stub [no-summary]
no area <area-id> stub [no-summary]
```

Parameter	Description	
<area-id>	The OSPF area that you are configuring as a stub area. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format. For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address in the format `A.B.C.D`.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
`no-summary`	Stops an ABR from sending summary link advertisements into the stub area.	

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# area 1 stub
```

Related Commands area default-cost

area virtual-link

This command configures a link between two backbone areas that are physically separated through other non-backbone areas.

In OSPF, all non-backbone areas must be connected to a backbone area. If the connection to the backbone is lost, the virtual link repairs the connection.

The **no** variant of this command removes the virtual link.

Syntax

```
area <area-id> virtual-link <ip-addr> [<auth-key>|<msg-key>]

no area <area-id>  virtual-link <ip-addr>[<auth-key>|<msg-key>]

area <area-id>  virtual-link <ip-addr> authentication
    [message-digest|null] [<auth-key>|<msg-key>]

no area <area-id>  virtual-link <ip-addr> authentication
    [message-digest|null] [<auth-key>|<msg-key>]

area <area-id>  virtual-link <ip-addr> [authentication]
    [dead-interval <1-65535>] [hello-interval <1-65535>]
    [retransmit-interval <1-3600>] [transmit-delay <1-3600>]

no area <area-id>  virtual-link <ip-addr>[authentication]
    [dead-interval] [hello-interval] [retransmit-interval] [transmit-
    delay]
```

Parameter	Description	
<area-id>	The area ID of the transit area that the virtual link passes through. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format.	
	<ip-addr>	OSPF Area ID expressed in IPv4 address format `A.B.C.D`.
	<0-4294967295>	OSPF Area ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area ID.	
<ip-address>	The OSPF router ID of the virtual link neighbor.	
<auth-key>	Specifies the password used for this virtual link. Use the format: **authentication-key** *<pswd-short>*	
	<pswd-short>	An 8 character password.
<msg-key>	Specifies a message digest key using the MD5 encryption algorithm. Use the following format: **message-digest-key** *<1-255>* md5 *<pswd-long>*	
	<1-255>	The key ID.
	<pswd-long>	Authentication password of 16 characters.
`authentication`	Enables authentication on this virtual link.	
`message-digest`	Use message-digest authentication.	

Parameter(cont.)	Description(cont.)	
`null`	Use null authentication to override password or message digest.	
`dead-interval`	If no packets are received from a particular neighbor for dead-interval seconds, the router considers that neighboring router as being off-line. Default: 40 seconds	
	<1-65535>	The number of seconds in the interval.
`hello-interval`	The interval the router waits before it sends a hello packet. Default: 10 seconds	
	<1-65535>	The number of seconds in the interval.
`retransmit-interval`	The interval the router waits before it retransmits a packet. Default: 5 seconds	
	<1-3600>	The number of seconds in the interval.
`transmit-delay`	The interval the router waits before it transmits a packet. Default: 1 seconds	
	<1-3600>	The number of seconds in the interval.

Mode Router Configuration

Usage You can configure virtual links between any two backbone routers that have an interface to a common non-backbone area. The protocol treats these two routers, joined by a virtual link, as if they were connected by an unnumbered point-to-point network. To configure a virtual link, you require:

- The transit area ID, i.e. the area ID of the non backbone area that the two backbone routers are both connected to.
- The corresponding virtual link neighbor's router ID. To see the router ID use the **show ip ospf** command.

Configure the **hello-interval** to be the same for all routers attached to a common network. A short **hello-interval** results in the router detecting topological changes faster but also an increase in the routing traffic.

The **retransmit-interval** is the expected round-trip delay between any two routers in a network. Set the value to be greater than the expected round-trip delay to avoid needless retransmissions.

The **transmit-delay** is the time taken to transmit a link state update packet on the interface. Before transmission, the link state advertisements in the update packet, are incremented by this amount. Set the **transmit-delay** to be greater than zero. Also, take into account the transmission and propagation delays for the interface.

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# area 1 virtual-link 10.10.11.50 hello 5
dead 10
```

Related Commands
area authentication
show ip ospf
show ip ospf virtual-links

auto-cost reference bandwidth

This command controls how OSPF calculates default metrics for the interface.

Use the **no** variant of this command to assign cost based only on the interface bandwidth.

Syntax

```
auto-cost reference-bandwidth <1-4294967>
no auto-cost reference-bandwidth
```

Parameter	Description
<1-4294967>	The reference bandwidth in terms of Mbits per second (Mbps).

Default 1000 Mbps

Usage By default, OSPF calculates the OSPF metric for an interface by dividing the reference bandwidth by the interface bandwidth. The default for the reference bandwidth is 1000 Mbps. As a result, if this default is used, there is very little difference between the metrics applied to interfaces of increasing bandwidth beyond 1000 Mbps.

The auto-cost command is used to alter this reference bandwidth in order to give a real difference between the metrics of high bandwidth links of differing bandwidths. In a network that has multiple links with high bandwidths, specify a larger reference bandwidth value to differentiate the costs on those links.

Cost is calculated by dividing the reference bandwidth (Mbps) by the layer 3 interface (Switched Virtual Interface (SVI), Loopback or Ethernet interface) bandwidth. Interface bandwidth may be altered by using the **bandwidth** command on page 42.16 as the SVI does not auto detect the bandwidth based on the speed of associated switch ports.

When the reference bandwidth calculation results in a cost integer greater than 1 but contains a fractional value (value after the decimal point), the result rounds down to the nearest integer. The following example shows how the cost is calculated.

The reference bandwidth is 1000 Mbps and the interface bandwidth is 7 Mbps.

Calculation = 1000/7

Calculation result = 142.85 (integer of 142, fractional value of 0.85)

Result after rounding down to the nearest integer = 142 (Interface cost is 142)

When the reference bandwidth calculation results in a cost less than 1, it is rounded up to the nearest integer which is 1. The following example shows how the cost is calculated.

The reference bandwidth is 1000 Mbps and the interface bandwidth is 10000 Mbps.

Calculation = 1000/10000

Calculation result = 0.1

Result after rounding up to the nearest integer = 1 (Interface cost is 1)

The auto-cost reference bandwidth value should be consistent across all OSPF routers in the OSPF process.

Note that using the **ip ospf cost** command on page 42.36 on a layer 3 interface will override the cost calculated by the reference bandwidth command.

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# auto-cost reference-bandwidth 1000
```

Related Commands **ip ospf cost**

bandwidth

Use this command to specify the maximum bandwidth to be used for each VLAN interface.

The bandwidth value is in bits. OSPF uses this to calculate metrics for the VLAN interface.

The **no** variant of this command removes any applied bandwidth value and replaces it with a value equal to the lowest port speed within that VLAN.

Syntax

```
bandwidth  <bandwidth-setting>
no bandwidth
```

Parameter	Description
<bandwidth-setting>	Sets the bandwidth for the interface. Enter a value in the range 1 to 10000000000 bits per second. Note that to avoid entering many zeros, you can add k, m, or g to internally add 3, 6 or 9 zeros to the number entered. For example entering 1k is the same as entering 1000.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# bandwidth 1000000
```

Related Commands

show running-config
show running-config access-list
show interface

capability opaque

This command enables opaque-LSAs. Opaque-LSAs are Type 9, 10 and 11 LSAs that deliver information used by external applications.

By default, opaque-LSAs are enabled.

Use the **no** variant of this command to disables opaque-LSAs.

Syntax

```
capability opaque
no capability opaque
```

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# no capability opaque
```

capability restart

This command enables OSPF Graceful Restart or restart signaling features. By default, this is enabled.

Use the **no** variant of this command to disable OSPF Graceful Restart and restart signaling features.

Syntax

```
capability restart [graceful|signaling]
no capability restart
```

Parameter	Description
`graceful`	Enable graceful OSPF restart.
`signaling`	Enable OSPF restart signaling.

Default Graceful restart

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# capability restart graceful
```

clear ip ospf process

This command clears and restarts the OSPF routing process. Specify the Process ID to clear one particular OSPF process. When no Process ID is specified, this command clears all running OSPF processes.

Syntax `clear ip ospf [<0-65535>] process`

Parameter	Description
<0-65535>	The Routing Process ID.

Mode Privileged Exec

Example

```
awplus# clear ip ospf process
```

compatible rfc1583

This command changes the method used to calculate summary route to the that specified in RFC 1583. By default, OSPF uses the method specified in RFC 2328.

RFC 1583 specifies a method for calculating the metric for summary routes based on the minimum metric of the component paths available. RFC 2328 specifies a method for calculating metrics based on maximum cost.

It is possible that some ABRs in an area might conform to RFC 1583 and others support RFC 2328, which could lead to incompatibility in their interoperation. This command addresses this issue by allowing you to selectively disable compatibility with RFC 2328.

Use the **no** variant of this command to disable RFC 1583 compatibility.

Syntax `compatible rfc1583`

`no compatible rfc1583`

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# compatible rfc1583
```

debug ospf events

This command enables OSPF debugging for OSPF event troubleshooting.

To enable all debugging options, specify **debug ospf event** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF debugging. Use this command without parameters to disable all the options.

Syntax

```
debug ospf events [abr] [asbr] [lsa] [nssa] [os] [router] [vlink]
no debug ospf events [abr] [asbr] [lsa] [nssa] [os] [router] [vlink]
```

Parameter	Description
`abr`	Shows ABR events.
`asbr`	Shows ASBR events.
`lsa`	Shows LSA events.
`nssa`	Shows NSSA events.
`os`	Shows OS interaction events.
`router`	Shows other router events.
`vlink`	Shows virtual link events.

Mode Privileged Exec and Global Configuration

Example

```
awplus# debug ospf events asbr lsa
```

Related Commands terminal monitor
undebug ospf events

debug ospf ifsm

This command specifies debugging options for OSPF Interface Finite State Machine (IFSM) troubleshooting.

To enable all debugging options, specify **debug ospf ifsm** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF IFSM debugging. Use this command without parameters to disable all the options.

Syntax

```
debug ospf ifsm [status] [events] [timers]
no debug ospf ifsm [status] [events] [timers]
```

Parameter	Description
`events`	Displays IFSM event information.
`status`	Displays IFSM status information.
`timers`	Displays IFSM timer information.

Mode Privileged Exec and Global Configuration

Example

```
awplus# no debug ospf ifsm events status
awplus# debug ospf ifsm status
awplus# debug ospf ifsm timers
```

Related Commands terminal monitor
undebug ospf ifsm

debug ospf lsa

This command enables debugging options for OSPF Link State Advertisements (LSA) troubleshooting. This displays information related to internal operations of LSAs.

To enable all debugging options, specify **debug ospf lsa** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF LSA debugging. Use this command without parameters to disable all the options.

Syntax

```
debug ospf lsa [flooding] [generate] [install] [maxage] [refresh]
no debug ospf lsa [flooding] [generate] [install] [maxage] [refresh]
```

Parameter	Description
`flooding`	Displays LSA flooding.
`generate`	Displays LSA generation.
`install`	Show LSA installation.
`maxage`	Shows maximum age of the LSA in seconds.
`refresh`	Displays LSA refresh.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# undebug ospf lsa refresh
```

Output **Figure 42-1: Example output from the debug ospf lsa command**

```
2002/05/09 14:08:11 OSPF: LSA[10.10.10.10:10.10.10.70]: instance(0x8139cd0)
created with Link State Update
2002/05/09 14:08:11 OSPF: RECV[LS-Upd]: From 10.10.10.70 via vlan5:10.10.10.50
(10.10.10.10 -> 224.0.0.5)
2002/05/09 14:12:33 OSPF: SEND[LS-Upd]: Begin send queue
2002/05/09 14:12:33 OSPF: SEND[LS-Upd]: # of LSAs 1, destination 224.0.0.5
2002/05/09 14:12:33 OSPF: SEND[LS-Upd]: End send queue
2002/05/09 14:12:33 OSPF: SEND[LS-Upd]: To 224.0.0.5 via vlan5:10.10.10.50
```

Related Commands terminal monitor
undebug ospf lsa

debug ospf nfsm

This command enables debugging options for OSPF Neighbor Finite State Machines (NFSMs).

To enable all debugging options, specify **debug ospf nfsm** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF NFSM debugging. Use this command without parameters to disable all the options.

Syntax

```
debug ospf nfsm [events] [status] [timers]
no debug ospf nfsm [events] [status] [timers]
```

Parameter	Description
`events`	Displays NFSM event information.
`status`	Displays NSFM status information.
`timers`	Displays NFSM timer information.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# debug ospf nfsm events
awplus# no debug ospf nfsm timers
awplus# undebug ospf nfsm events
```

Related Commands **terminal monitor**
undebug ospf nfsm

debug ospf nsm

This command enables debugging options for the OSPF Network Service Module.

To enable both debugging options, specify **debug ospf nsm** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF NSM debugging. Use this command without parameters to disable both options.

Syntax

```
debug ospf nsm [interface] [redistribute]
no debug ospf nsm [interface] [redistribute]
```

Parameter	Description
`interface`	Specify NSM interface information.
`redistribute`	Specify NSM redistribute information.

Mode Priviledged Exec and Global Configuration

Examples

```
awplus# debug ospf nsm interface
awplus# no debug ospf nsm redistribute
awplus# undebug ospf nsm interface
```

Related Commands **terminal monitor**
undebug ospf nsm

debug ospf packet

This command enables debugging options for OSPF packets.

To enable all debugging options, specify **debug ospf packet** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF packet debugging. Use this command without parameters to disable all options.

Syntax

```
debug ospf packet [dd] [detail] [hello] [ls-ack] [ls-request]
    [ls-update] [recv] [send]

no debug ospf packet [dd] [detail] [hello] [ls-ack] [ls-request]
    [ls-update] [recv] [send]
```

Parameter	Description
`dd`	Specifies debugging for OSPF database descriptions.
`detail`	Sets the debug option to detailed information.
`hello`	Specifies debugging for OSPF hello packets.
`ls-ack`	Specifies debugging for OSPF link state acknowledgments.
`ls-request`	Specifies debugging for OSPF link state requests.
`ls-update`	Specifies debugging for OSPF link state updates.
`recv`	Specifies the debug option set for received packets.
`send`	Specifies the debug option set for sent packets.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# debug ospf packet detail
awplus# debug ospf packet dd send detail
awplus# no debug ospf packet ls-request recv detail
awplus# undebug ospf packet ls-request recv detail
```

Related Commands **terminal monitor**
undebug ospf packet

debug ospf route

This command enables debugging of route calculation. Use this command without parameters to turn on all the options.

To enable all debugging options, specify **debug ospf route** with no additional parameters.

The **no** and **undebug** variant of this command disable OSPF route debugging. Use this command without parameters to disable all options.

Syntax

```
debug ospf route [ase] [ia] [install] [spf]
no debug ospf route [ase] [ia] [install] [spf]
```

Parameter	Description
`ia`	Specifies the debugging of Inter-Area route calculation.
`ase`	Specifies the debugging of external route calculation.
`install`	Specifies the debugging of route installation.
`spf`	Specifies the debugging of SPF calculation.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# debug ospf route
awplus# no debug ospf route ia
awplus# debug ospf route install
awplus# undebug ospf route install
```

Related Commands **terminal monitor**
undebug ospf route

default-information originate (OSPF)

This command creates a default external route into an OSPF routing domain.

When you use the **default-information originate** command to redistribute routes into an OSPF routing domain, then the system acts like an Autonomous System Boundary Router (ASBR). An ASBR does not by default, generate a default route into the OSPF routing domain.

When using this command, also specify the **route-map** <***route-map***> option to avoid a dependency on the default network in the routing table.

The **metric-type** is an external link type associated with the default route advertised into the OSPF routing domain. The value of the external route could be either Type 1 or 2. The default is Type 2.

The **no** variant of this command disables this feature.

Syntax

```
default-information originate [always] [metric <metric>]
    [metric-type <1-2>] [route-map <route-map>]

no default-information originate [always] [metric] [metric-type]
    [route-map]
```

Parameter	Description
`always`	Used to advertise the default route regardless of whether there is a default route.
`metric <metric>`	The metric value used in creating the default route. Enter a value in the range 0 to 16777214. The default metric value is 10. The value used is specific to the protocol.
`metric-type <1-2>`	External metric type for default routes, either OSPF External Type 1 or Type 2 metrics. Enter the value 1 or 2.
`route-map`	Specifies to use a specific route-map.
`<route-map>`	The route-map name. It is a string comprised of any characters, numbers or symbols.

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# default-information originate
                       always metric 23 metric-type 2
                       route-map myinfo
```

Related Commands route-map

default-metric (OSPF)

This command sets default metric values for the OSPF routing protocol.

The **no** variant of this command returns OSPF to using built-in, automatic metric translations, as appropriate for each routing protocol.

Syntax

```
default-metric <1-16777214>
no default-metric [<1-16777214>]
```

Parameter	Description
<1-16777214>	Default metric value appropriate for the specified routing protocol.

Mode Router Configuration

Usage A default metric facilitates redistributing routes even with incompatible metrics. If the metrics do not convert, the default metric provides an alternative and enables the redistribution to continue. The effect of this command is that OSPF will use the same metric value for **all** redistributed routes. Use this command in conjunction with the **redistribute (OSPF)** command.

Examples

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# default-metric 100

awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# no default-metric
```

Related commands **redistribute (OSPF)**

distance (OSPF)

This command sets the administrative distance for OSPF routes based on the route type. Your switch uses this value to select between two or more routes to the same destination from two different routing protocols. The route with the smallest administrative distance value is added to the Forwarding Information Base (FIB). See **"Administrative Distance" on page 35.6** for more information.

Use the command **distance ospf** to set the distance for an entire category of OSPF routes, rather than the specific routes that pass an access list.

Use the command **distance <1-255>**, with no other parameter, to set the same distance for all OSPF route types.

The **no** variant of this command sets the administrative distance for all OSPF routes to the default of 110.

Syntax

```
distance <1-255>

distance ospf {external <1-255>|inter-area <1-255>|
   intra-area <1-255>}

no distance {ospf|<1-255>}
```

Parameter	Description
`<1-255>`	Specify the Administrative Distance value for OSPF routes.
`external`	Sets the distance for routes from other routing domains, learned by redistribution. Specify an OSPF external distance in the range `<1-255>`.
`inter-area`	Sets the distance for all routes from one area to another area. Specify an OSPF inter-area distance in the range `<1-255>`.
`intra-area`	Sets the distance for all routes within an area. Specify an OSPF intra-area distance in the range `<1-255>`.

Default The default OSPF administrative distance is 110. The default Administrative Distance for each type of route (intra, inter, or external) is 110.

Mode Router Configuration

Usage The administrative distance rates the trustworthiness of a routing information source. The distance could be any integer from 0 to 255. A higher distance value indicates a lower trust rating. For example, an administrative distance of 255 indicates that the routing information source cannot be trusted and should be ignored.

Use this command to set the distance for an entire group of routes, rather than a specific route that passes an access list.

Examples To set the following administrative distances for route types in OSPF 100:

- 20 for inter-area routes
- 10 for intra-area routes
- 40 for external routes

use the commands:

```
awplus(config)# router ospf 100
awplus(config-router)# distance ospf inter-area 20 intra-area
10 external 40
```

To set the administrative distance for all routes in OSPF 100 back to the default of 110, use the commands:

```
awplus(config)# router ospf 100
awplus(config-router)# no distance ospf
```

distribute-list (OSPF)

Use this command to apply filtering to the transfer of routing information between OSPF and the IP route table. You can apply filtering in either direction, from OSPF to the IP route table using an **in** distribute-list, or from the IP route table to OSPF using an **out** distribute-list.

The effect of an **in** filter is that some route information that OSPF has learned from LSA updates will not be installed into the IP route table. The effect of an **out** filter is that some route information that could be redistributed to OSPF will not be redistributed to OSPF. See the **Usage** section below for the distinction between the **in** and **out** distribute-lists.

The entities that are used to perform filtering are ACLs or route-maps, which match on certain attributes in the routes that are being transferred. See **Access Control Lists Introduction** and **Route Map Configuration** chapters to define ACLs and route-maps.

The **no** variant of this command removes the configured distribute-list command entry.

Syntax

```
distribute-list {<access-list-name>|route-map <route-map-name>} in
distribute-list <access-list-name> out {connected|rip|static}
no distribute-list <access-list-name> in
no distribute-list <access-list-name> out {connected|rip|static}
```

Parameter	Description
<access-list-name>	Specifies the name of the access list. The access list defines which networks are received and which are suppressed.
`in`	Indicates that this applies to incoming advertised routes.
`out`	Indicates that this applies to outgoing advertised routes.
<route-map-name>	The name of the route-map that the distribute-list applies. This defines which networks are installed in the IP route table and which networks are filtered from the IP route table.
`connected`	Specify the redistribution of connected routes.
`rip`	Specify the redistribution of RIP routes.
`static`	Specify the redistribution of static routes.

Mode Router Configuration

Usage There are **in** and **out** distribute-lists, which carry out different route filtering activities:

- The **in** distribute list is applied to the process of installing OSPF routes into the IP route table. The SPF calculation generate a set of routes calculated from the LSA database. By default, all of these routes become OSPF's candidate routes for inclusion into the IP route table.

- An **in** distribute-list can be used to control whether or not certain routes generated by the SPF calculation are included into the set of candidates for inclusion into the IP route table. Those routes that match **deny** entries in the distribute-list will not be considered for inclusion into the IP route table.
- The **out** distribute-list applies the process of redistributing non-OSPF routes into OSPF. If OSPF redistribution is configured, and an **out** distribute-list is also configured, then routes that match deny entries in the distribute-list will not be redistributed into OSPF.

Examples The following example shows the installation of OSPF routes into the IP route table with route map `mymap1` applied, which will process routes that have been tagged `100`:

```
awplus# configure terminal
awplus(config)# route-map mymap1 permit 10
awplus(config-route-map)# match tag 100
awplus(config-route-map)# exit
awplus(config)# router ospf 100
awplus(config-router)# distribute-list route-map mymap1 in
```

Use the following commands to configure a route-map to specifically prevent OSPF from offering `192.168.1.0/24` as a candidate for inclusion into the IP route table:

```
awplus# configure terminal
awplus(config)# ip prefix-list 100 seq 5 permit
192.168.1.0/24
awplus(config)# route-map 100 deny 10
awplus(config-route-map)# match ip address prefix-list 100
awplus(config-route-map)# exit
awplus(config)# route-map 100 permit 20
awplus(config-router)# router ospf 1
awplus(config-router)# distribute-list route-map 100 in
```

Related Commands **match interface**
redistribute (OSPF)
route-map

enable db-summary-opt

This command enables OSPF database summary list optimization.

The **no** variant of this command disables database summary list optimization.

Syntax

```
enable db-summary-opt
no enable db-summary-opt
```

Default The default setting is disabled.

Mode Router Configuration

Usage When this feature is enabled, the database exchange process is optimized by removing the LSA from the database summary list for the neighbor, if the LSA instance in the database summary list is the same as, or less recent than, the listed LSA in the database description packet received from the neighbor.

Examples To enable OSPF database summary list optimization, use the commands:

```
awplus# configure terminal
awplus(config)# router ospf
awplus(config-router)# enable db-summary-opt
```

To disable OSPF database summary list optimization, use the commands:

```
awplus# configure terminal
awplus(config)# router ospf
awplus(config-router)# no enable db-summary-opt
```

Validation Commands **show running-config**

host area

This command configures a stub host entry belonging to a particular area. You can use this command to advertise specific host routes in the router-LSA as stub link. Since stub host belongs to the specified router, specifying cost is optional.

The **no** variant of this command removes the host area configuration.

Syntax

```
host <ip-address> area <area-id> [cost <0-65535>]
no host <ip-address> area <area-id> [cost <0-65535>]
```

Parameter	Description
<ip-address>	The IPv4 address of the host, in dotted decimal notation.
<area-id>	The OSPF area ID of the transit area that configuring the stub host entry for. Use one of the following formats: ■ dotted decimal format, e.g. 0.0.1.2. ■ normal decimal format in the range <0-4294967295>, e.g. 258.
`cost <0-65535>`	The cost for the stub host entry.

Default By default, no host entry is configured.

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# host 172.16.10.100 area 1
awplus(config-router)# host 172.16.10.101 area 2 cost 10
```

ip ospf authentication

This command sets the authentication method used when sending and receiving OSPF packets on the current VLAN interface. The default is to use no authentication. If no authentication method is specified in this command, then plain text authentication will be used.

The **no** variant of this command disables the authentication.

Syntax

```
ip ospf [<ip-address>] authentication [message-digest|null]
no ip ospf [<ip-address>] authentication
```

Parameter	Description
<ip-address>	The IP address of the interface.
`message-digest`	Use the message digest authentication.
`null`	Use no authentication. It overrides password or message-digest authentication of the interface.

Mode Interface Configuration for a VLAN interface.

Usage Use the **ip ospf authentication** command on page 42.34 to specify a Simple Text password. Use the **ip ospf message-digest-key** command on page 42.41 to specify MD5 password.

Example In this example, VLAN interface `vlan2` is configured to have no authentication. This will override any text or MD5 authentication configured on this interface.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf authentication null
```

Related Commands
ip ospf authentication-key
area authentication
ip ospf message-digest-key

ip ospf authentication-key

This command specifies an OSPF authentication password for the neighboring routers.

The **no** variant of this command removes the OSPF authentication password.

Syntax

```
ip ospf [<ip-address>] authentication-key <pswd-long>
no ip ospf [<ip-address>] authentication-key
```

Parameter	Description
<ip-address>	The IPv4 address of the interface, in dotted decimal notation.
<pswd-long>	Specifies the authentication password. The string by the end of line will be used.

Default By default, an authentication password is not specified.

Mode Interface Configuration for a VLAN interface.

Usage This command creates a password (key) that is inserted into the OSPF header when AlliedWare Plus™ software originates routing protocol packets. Assign a separate password to each network for different VLAN interfaces. All neighboring routers on the same network with the same password exchange OSPF routing data.

The key can be used only when authentication is enabled for an area. Use the **area authentication** command to enable authentication.

Simple password authentication allows a password to be configured for each area. Configure the routers in the same routing domain with the same password.

Example In the following example, an authentication key test is created on VLAN interface `vlan2` in area 0. Note that first authentication is enabled for area 0.

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# network 10.10.10.0/24 area 0
awplus(config-router)# area 0 authentication
awplus(config-router)# exit
awplus(config)# interface vlan2
awplus(config-if)# ip ospf 3.3.3.3 authentication-key test
```

Related Commands **area authentication**
ip ospf authentication

ip ospf cost

This command explicitly specifies the cost of the link-state metric in a router-LSA.

The **no** variant of this command resets the VLAN interface cost to the default.

Syntax

```
ip ospf [<ip-address>] cost <1-65535>
no ip ospf [<ip-address>] cost
```

Parameter	Description
<ip-address>	The IPv4 address of the interface, in dotted decimal notation.
<1-65535>	The link-state metric.

Default By default there is no static value set and the OSPF cost is automatically calculated by using the **auto-cost reference bandwidth** command on page 42.14.

Mode Interface Configuration for a VLAN interface.

Usage This command explicitly sets a user specified cost of sending packets out the interface. Using this command overrides the cost value calculated automatically with the auto-cost reference bandwidth feature.

The interface cost indicates the overhead required to send packets across a certain VLAN interface. This cost is stated in the Router-LSA's link. Typically, the cost is inversely proportional to the bandwidth of an interface. By default, the cost of a VLAN interface is calculated according to the following formula:

reference bandwidth/interface bandwidth

To set the VLAN interface cost manually, use this command.

Example The following example shows setting ospf cost to 10 on VLAN interface `vlan25` for IP address 10.10.10.50

```
awplus# configure terminal
awplus(config)# interface vlan25
awplus(config-if)# ip ospf 10.10.10.50 cost 10
```

Related Commands

show ip ospf interface
auto-cost reference bandwidth

ip ospf database-filter

This command turns on the LSA database-filter for a particular VLAN interface.

The **no** variant of this command turns off the LSA database-filter.

Syntax

```
ip ospf [<ip-address>] database-filter all out
no ip ospf [<ip-address>] database-filter
```

Parameter	Description
<ip-address>	The IPv4 address of the interface, in dotted decimal notation.

Default By default, all outgoing LSAs are flooded to the interface.

Mode Interface Configuration for a VLAN interface.

Usage OSPF floods new LSAs over all interfaces in an area, except the interface on which the LSA arrives. This redundancy ensures robust flooding. However, too much redundancy can waste bandwidth and might lead to excessive link and CPU usage in certain topologies, resulting in destabilizing the network. To avoid this, use the **ip ospf database-filter** command to block flooding of LSAs over specified interfaces.

Example

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if# ip ospf database-filter all out
```

ip ospf dead-interval

This command sets the interval during which no hello packets are received and after which a neighbor is declared dead.

The dead-interval is the amount of time that OSPF waits to receive an OSPF hello packet from the neighbor before declaring the neighbor is down. This value is advertised in the router's hello packets. It must be a multiple of the hello-interval and be the same for all routers on a specific network.

The **no** variant of this command returns the interval to the default of 40 seconds. If you have configured this command specifying the IP address of the interface and want to remove the configuration, specify the IP address (**no ip ospf** `<ip-address>` **dead-interval**).

Syntax

```
ip ospf [<ip-address>] dead-interval <1-65535>
no ip ospf [<ip-address>] dead-interval
```

Parameter	Description
`<ip-address>`	The IPv4 address of the interface, in dotted decimal notation.
`<1-65545>`	The interval in seconds. Default: **40**

Mode Interface Configuration for a VLAN interface.

Example The following example shows configuring the dead-interval to 10 seconds on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf dead-interval 10
```

Related Commands

ip ospf hello-interval
show ip ospf interface

ip ospf disable all

This command completely disables OSPF packet processing on a VLAN interface. It overrides the **network area** command and disables the processing of packets on the specific interface.

Use the **no** variant of this command to restore OSPF packet processing on a selected interface.

Syntax

```
ip ospf disable all
no ip ospf disable all
```

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf disable all
```

ip ospf hello-interval

This command specifies the interval between hello packets.

The hello-interval is advertised in the hello packets. Configure the same hello-interval for all routers on a specific network. A shorter hello interval ensures faster detection of topological changes, but results in more routing traffic.

The **no** variant of this command returns the interval to the default of 10 seconds.

Syntax

```
ip ospf [<ip-address>] hello-interval <1-65535>
no ip ospf [<ip-address>] hello-interval
```

Parameter	Description
<ip-address>	The IP address of the interface, in dotted decimal notation.
<1-65535>	The interval in seconds. Default: **10**

Default The default interval is 10 seconds.

Mode Interface Configuration for a VLAN interface.

Example The following example shows setting the hello-interval to 3 seconds on VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf hello-interval 3
```

Related Commands

ip ospf dead-interval
show ip ospf interface

ip ospf message-digest-key

This command registers an MD5 key for OSPF MD5 authentication.

Message Digest Authentication is a cryptographic authentication. A key (password) and key-id are configured on each router. The router uses an algorithm based on the OSPF packet, the key, and the key-id to generate a **message digest** that gets appended to the packet.

The **no** variant of this command removes the MD5 key.

Syntax

```
ip ospf [<ip-address>] message-digest-key <key-id> md5 <pswd-long>
no ip ospf [<ip-address>] message-digest-key <key-id>
```

Parameter	Description
<ip-address>	The IPv4 address of the interface, in dotted decimal notation.
<key-id>	A key ID number specified as an integer between 1 and 255.
md5	Use the MD5 algorithm.
<pswd-long>	The OSPF password. This is a string of 1 to 16 characters including spaces.

Default By default, there is no MD5 key registered.

Mode Interface Configuration for a VLAN interface.

Usage Use this command for uninterrupted transitions between passwords. It allows you to add a new key without having to delete the existing key. While multiple keys exist, all OSPF packets will be transmitted in duplicate; one copy of the packet will be transmitted for each of the current keys. This is helpful for administrators who want to change the OSPF password without disrupting communication. The system begins a rollover process until all the neighbors have adopted the new password. This allows neighboring routers to continue communication while the network administrator is updating them with a new password. The router will stop sending duplicate packets once it detects that all of its neighbors have adopted the new password.

Maintain only one password per interface, removing the old password whenever you add a new one. This will prevent the local system from continuing to communicate with the system that is using the old password. Removing the old password also reduces overhead during rollover. All neighboring routers on the same network must have the same password value to enable exchange of OSPF routing data.

Examples The following example shows OSPF authentication on the VLAN interface `vlan5` when IP address has not been specified.

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ip ospf authentication message-digest
awplus(config-if)# ip ospf message-digest-key 1 md5 yourpass
```

The following example shows configuring OSPF authentication on the VLAN interface `vlan2` for the IP address `1.1.1.1`. (If the interface has two IP addresses assigned--`1.1.1.1` & `2.2.2.2`, OSPF authentication will be enabled only for the IP address `1.1.1.1`).

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf 1.1.1.1 authentication message-
digest
awplus(config-if)# ip ospf 1.1.1.1 message-digest-key 2 md5
yourpass
```

ip ospf mtu

This command sets the MTU size for OSPF. Whenever OSPF constructs packets, it uses VLAN interface MTU size as Maximum IP packet size. This command forces OSPF to use the specified value, overriding the actual VLAN interface MTU size.

Use the **no** variant of this command to return the MTU size to the default.

Syntax

```
ip ospf mtu <576-65535>
no ip ospf mtu
```

Default By default, OSPF uses interface MTU derived from the VLAN interface.

Mode Interface Configuration for a VLAN interface.

Usage This command allows an administrator to configure the MTU size recognized by the OSPF protocol. It does not configure the MTU settings on the VLAN interface. OSPF will not recognize MTU size configuration changes made to the kernel until the MTU size is updated through the CLI.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf mtu 1480
```

ip ospf mtu-ignore

Use this command to configure OSPF so that OSPF does not check the MTU size during DD (Database Description) exchange.

Use the **no** variant of this command to make sure that OSPF checks the MTU size during DD exchange.

Syntax

```
ip ospf [<ip-address>] mtu-ignore
no ip ospf [<ip-address>] mtu-ignore
```

Parameter	Description
<ip-address>	IPv4 address of the interface, in dotted decimal notation.

Mode Interface Configuration for a VLAN interface.

Usage By default, during the DD exchange process, OSPF checks the MTU size described in the DD packets received from the neighbor. If the MTU size does not match the interface MTU, the neighbor adjacency is not established. Using this command makes OSPF ignore this check and allows establishing of adjacency regardless of MTU size in the DD packet.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf mtu-ignore
```

ip ospf network

This command configures the OSPF network type to a type different from the default for the particular VLAN interface.

The **no** variant of this command returns the network type to the default for the particular VLAN interface.

Syntax

```
ip ospf network [broadcast|non-broadcast|point-to-point|
   point-to-multipoint]

no ip ospf network
```

Parameter	Description
`broadcast`	Sets the network type to broadcast.
`non-broadcast`	Sets the network type to NBMA.
`point-to-multipoint`	Sets the network type to point-to-multipoint.
`point-to-point`	Sets the network type to point-to-point.

Default The default is the `broadcast` OSPF network type for a VLAN interface.

Mode Interface Configuration for a VLAN interface.

Usage This command forces the interface network type to the specified type. Depending on the network type, OSPF changes the behavior of the packet transmission and the link description in LSAs.

Example The following example shows setting the network type to `point-to-point` on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf network point-to-point
```

ip ospf priority

This command sets the router priority, which is a parameter used in the election of the designated router for the network.

The **no** variant of this command returns the router priority to the default of 1.

Syntax

```
ip ospf [<ip-address>] priority <priority>
no ip ospf [<ip-address>] priority
```

Parameter	Description
<ip-address>	The IP address of the interface.
<priority>	*<0-255>* Specifies the Router Priority of the interface.

Default The router priority for an interface is set to 1 by default.

Mode Interface Configuration for a VLAN interface.

Usage Set the priority to help determine the OSPF Designated Router (DR) for a network. If two routers attempt to become the DR, the router with the higher router priority becomes the DR. If the router priority is the same for two routers, the router with the higher router ID takes precedence.

Only routers with nonzero router priority values are eligible to become the designated or backup designated router.

Configure router priority for multi-access networks only and not for point-to-point networks.

Example The following example shows setting the OSPF priority value to 3 on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf priority 3
```

Related Commands ip ospf network

ip ospf resync-timeout

Use this command to set the interval after which adjacency is reset if out-of-band resynchronization has not occurred. The interval period starts from the time a restart signal is received from a neighbor.

Use the **no** variant of this command to return to the default.

Syntax

```
ip ospf [<ip-address>] resync-timeout <1-65535>
no ip ospf [<ip-address>] resync-timeout
```

Parameter	Description
<ip-address>	The IP address of the interface.
<1-65535>	Specifies the resynchronization timeout value of the interface in seconds.

Mode Interface Configuration for a VLAN interface.

Example The following example shows setting the OSPF resynchronization timeout value to 65 seconds on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf resync-timeout 65
```

ip ospf retransmit-interval

Use this command to specify the time between link-state advertisement (LSA) retransmissions for adjacencies belonging to the interface.

Use the **no** variant of this command to return to the default of 5 seconds.

Syntax

```
ip ospf [<ip-address>] retransmit-interval <1-65535>
no ip ospf [<ip-address>] retransmit-interval
```

Parameter	Description
<ip-address>	The IP address of the interface.
<1-65535>	Specifies the interval in seconds.

Default The default interval is 5 seconds.

Mode Interface Configuration for a VLAN interface.

Usage After sending an LSA to a neighbor, the router keeps the LSA until it receives an acknowledgment. In case the router does not receive an acknowledgment during the set time (the retransmit interval value) it retransmits the LSA. Set the retransmission interval value conservatively to avoid needless retransmission. The interval should be greater than the expected round-trip delay between two routers.

Example The following example shows setting the `ospf retransmit interval` to 6 seconds on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf retransmit-interval 6
```

ip ospf transmit-delay

Use this command to set the estimated time it takes to transmit a link-state-update packet on the VLAN interface.

Use the **no** variant of this command to return to the default of 1 second.

Syntax

```
ip ospf [<ip-address>] transmit-delay <1-65535>
no ip ospf [<ip-address>] transmit-delay
```

Parameter	Description
<ip-address>	The IP address of the VLAN interface.
<1-65535>	Specifies the time, in seconds, to transmit a link-state update.

Default The default interval is 1 second.

Mode Interface Configuration for a VLAN interface.

Usage The transmit delay value adds a specified time to the age field of an update. If the delay is not added, the time in which the LSA transmits over the link is not considered. This command is especially useful for low speed links. Add transmission and propagation delays when setting the transmit delay value.

Example The following example shows setting the OSPF transmit delay time to 3 seconds on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip ospf transmit-delay 3
```

max-concurrent-dd

Use this command to set the limit for the number of Database Descriptors (DD) that can be processed concurrently.

Use the **no** variant of this command to reset the limit for the number of Database Descriptors (DD) that can be processed concurrently.

Syntax

```
max-concurrent-dd <1-65535>
no max-concurrent-dd
```

Parameter	Description
<1-65535>	Specify the number of DD processes.

Mode Router Configuration

Usage This command is useful when a router's performance is affected from simultaneously bringing up several OSPF adjacencies. This command limits the maximum number of DD exchanges that can occur concurrently per OSPF instance, thus allowing for all of the adjacencies to come up.

Example The following example sets the max-concurrent-dd value to 4, so that only 4 DD exchanges will be processed at a time.

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router# max-concurrent-dd 4
```

maximum-area

Use this command to set the maximum number of OSPF areas.

Use the **no** variant of this command to set the maximum number of OSPF areas to the default.

Syntax

```
maximum-area <1-4294967294>
no maximum-area
```

Parameter	Description
<1-4294967294>	Specify the maximum number of OSPF areas.

Default The default for the maximum number of OSPF areas is 4294967294.

Mode Router Configuration

Usage Use this command in router OSPF mode to specify the maximum number of OSPF areas.

Examples The following example sets the maximum number of OSPF areas to 2:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# maximum-area 2
```

The following example removes the maximum number of OSPF areas and resets to default:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# no maximum-area
```

neighbor (OSPF)

Use this command to inform the router of other neighboring routers that are connected to the same NBMA network.

Use the **no** variant of this command to remove a configuration.

Syntax

```
neighbor <ip-address> [<cost>]{<priority>|<poll-interval>}
no neighbor <ip-address> [<cost>]{<priority>|<poll-interval>}
```

Parameter	Description
<ip-address>	Specifies the interface IP address of the neighbor.
<priority>	`priority` *<0-255>* Specifies the router priority value of the non-broadcast neighbor associated with the specified IP address. The default is 0. This keyword does not apply to point-to-multipoint interfaces.
<poll-interval>	`poll-interval` *<1-2147483647>* Dead neighbor polling interval in seconds. It is recommended to set this value much higher than the hello interval. The default is 120 seconds.
<cost>	`cost` *<1-65535>* Specifies the link-state metric to this neighbor.

Mode Router Configuration

Usage To configure a neighbor on an NBMA network manually, use the `neighbor` command and include one neighbor entry for each known nonbroadcast network neighbor. The IP address used in this command is the neighbor's primary IP address on the interface where that neighbor connects to the NBMA network.

The poll interval is the reduced rate at which routers continue to send hello packets, when a neighboring router has become inactive. Set the poll interval to be much larger than hello interval.

Examples This example shows a neighbor configured with a priority value, poll interval time, and cost.

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# neighbor 1.2.3.4 priority 1 poll-
interval 90
awplus(config-router)# neighbor 1.2.3.4 cost 15
```

network area

Use this command to enable OSPF routing with a specified Area ID on any interfaces with IP addresses that match the specified network address.

Use the **no** variant of this command to disable OSPF routing on the interfaces.

Syntax

```
network <network-address> area <area-id>
no network <network-address> area <area-id>
```

Parameter	Description
<network-address>	{*<ip-network/m>*|*<ip-addr> <reverse-mask>*}
<ip-network/m>	IP address of the network, entered in the form A.B.C.D/M. Dotted decimal notation followed by a forward slash, and then the subnet mask length.
<ip-addr> <reverse-mask>	IPv4 network address, entered in the form A.B.C.D, followed by the mask. Enter the mask as a wildcard, or reverse, mask (e.g. 0.0.0.255). Note that the switch displays the mask as a subnet mask in the running configuration.
<area-id>	{*<ip-addr>*|*<0-4294967295>*}
<ip-addr>	OSPF Area ID in IPv4 address format, in the form A.B.C.D.
<0-4294967295>	OSPF Area ID as 4 octets unsigned integer value.

Default No **network area** is configured by default.

Mode Router Configuration

Usage OSPF routing can be enabled per IPv4 subnet. The network address can be defined using either the prefix length or a wild card mask. A wild card mask is comprised of consecutive `0's` as network bits and consecutive `1's` as host bits.

Examples The following commands show the use of the **network area** command with OSPF multiple instance support disabled:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# network 10.0.0.0/8 area 3
awplus(config-router)# network 10.0.0.0/8 area 1.1.1.1
```

The following commands disable OSPF routing with Area ID 3 on all interfaces:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# no network 10.0.0.0/8 area3
```

ospf abr-type

Use this command to set an OSPF Area Border Router (ABR) type.

Use the **no** variant of this command to revert the ABR type to the default setting (Cisco).

Syntax

```
ospf abr-type {cisco|ibm|standard}
no ospf abr-type {cisco|ibm|standard}
```

Parameter	Description
`cisco`	Specifies an alternative ABR using Cisco implementation (RFC 3509). This is the default ABR type.
`ibm`	Specifies an alternative ABR using IBM implementation (RFC 3509).
`standard`	Specifies a standard behavior ABR (RFC 2328).

Default ABR type `Cisco`

Mode Router Configuration

Usage Specifying the ABR type allows better interoperation between different implementations. This command is specially useful in a multi-vendor environment. The different ABR types are:

- Cisco ABR Type: By this definition, a router is considered an ABR if it has more than one area actively attached and one of them is the backbone area.
- IBM ABR Type: By this definition, a router is considered an ABR if it has more than one area actively attached and the backbone area is configured. In this case the configured backbone need not be actively connected.
- Standard ABR Type: By this definition, a router is considered an ABR if it has more than one area actively attached to it.

Example

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# ospf abr-type ibm
```

ospf restart grace-period

Use this command to configure the grace-period for restarting OSPF routing.

Use the **no** variant of this command to revert to the default grace-period.

Syntax

```
ospf restart grace-period <1-1800>
no ospf restart grace-period
```

Parameter	Description
<1-1800>	Specifies the grace period in seconds.

Default In the AlliedWare Plus™ OSPF implementation, the default OSPF grace-period is 180 seconds.

Mode Global Configuration

Usage Use this command to enable the OSPF Graceful Restart feature and set the restart grace-period. Changes from the default restart grace-period are displayed in the running-config. The restart grace-period is not displayed in the running-config if it has been reset to the default using the **no** variant of this command.

When a master failover happens on a VCStack, the OSPF grace-period will be the longest period between the default value (180 seconds is the default OSPF grace-period) and the configured OSPF grace-period value from this command. So the configured OSPF grace-period value will not be used for a VCStack master failover if it is shorter than the default OSPF grace-period.

Example To set the OSPF restart grace-period to 250 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ospf restart grace-period 250
```

To reset the OSPF restart grace-period to the default (180 seconds), use the commands:

```
awplus# configure terminal
awplus(config)# no ospf restart grace-period
```

Validation Commands **show running-config**

Related Commands **ospf restart helper**
restart ospf graceful

ospf restart helper

Use this command to configure the **helper** behavior for the OSPF Graceful Restart feature.

Use the **no** variant of this command to revert to the default grace-period.

Syntax

```
ospf restart helper {max-grace-period <grace-period>|only-reload|
   only-upgrade}

ospf restart helper {never router-id <router-id>}

no ospf restart helper [max-grace-period]
```

Parameter	Description
`max-grace-period`	Specify help if received grace-period is less than a specified value.
<grace-period>	Maximum grace period accepted in seconds in range *<1-1800>*.
`never`	Specify the local policy to never to act as a helper for this feature.
`only-reload`	Specify help only on software reloads not software upgrades.
`only-upgrade`	Specify help only on software upgrades not software reloads.
`router-id`	Enter the router-id keyword to specify the OSPF Router ID that is never to act as a helper for the OSPF Graceful Restart feature.
<router-id>	*<A.B.C.D>* Specify the OSPF Router ID in dotted decimal format A.B.C.D

Default In the AlliedWare Plus™ OSPF implementation, the default OSPF grace-period is 180 seconds.

Mode Global Configuration

Usage The **ospf restart helper** command requires at least one parameter, but you may use more than one in the same command (excluding parameter **never**).

The **no** version of this command turns off the OSPF restart helper, while the **no ospf restart helper max-grace-period** command resets the max-grace-period, rather than the helper policy itself.

Example

```
awplus# configure terminal
awplus(config)# ospf restart helper only-reload

awplus# configure terminal
awplus(config)# ospf restart helper never router-id 10.10.10.1
```

```
awplus# configure terminal
awplus(config)# no ospf restart helper max-grace-period
```

Related Commands **ospf restart grace-period**
restart ospf graceful

ospf router-id

Use this command to specify a router ID for the OSPF process.

Use the **no** variant of this command to disable this function.

Syntax

```
ospf router-id <ip-address>
no ospf router-id
```

Parameter	Description
<ip-address>	Specifies the router ID in IPv4 address format.

Mode Router Configuration

Usage Configure each router with a unique router-id. In an OSPF router process that has active neighbors, a new router-id takes effect at the next reload or when you restart OSPF manually.

Example The following example shows a specified router ID 2.3.4.5.

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# ospf router-id 2.3.4.5
```

Related Commands show ip ospf

overflow database

Use this command to limit the maximum number of Link State Advertisements (LSAs) that can be supported by the current OSPF instance.

Use the **no** variant of this command to have no limit on the maximum number of LSAs.

Syntax

```
overflow database <0-4294967294> {hard|soft}
no overflow database
```

Parameter	Description
<0-4294967294>	The maximum number of LSAs.
`hard`	Shutdown occurs if the number of LSAs exceeds the specified value.
`soft`	Warning message appears if the number of LSAs exceeds the specified value.

Mode Router Configuration

Usage Use **hard** with this command if a shutdown is required if the number of LSAs exceeds the specified number. Use **soft** with this command if a shutdown is not required, but a warning message is required, if the number of LSAs exceeds the specified number.

Example The following example shows setting the database overflow to 500, and a shutdown to occur, if the number of LSAs exceeds 500.

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# overflow database 500 hard
```

overflow database external

Use this command to configure the size of the external database and the time the router waits before it tries to exit the overflow state.

Use the **no** variant of this command to revert to default.

Syntax

```
overflow database external <max-lsas> <recover-time>
no overflow database external
```

Parameter	Description
<max-lsas>	*<0-2147483647>* The maximum number of Link State Advertisements (LSAs). Note that this value should be the same on all routers in the AS.
<recover-time>	*<0-65535>* the number of seconds the router waits before trying to exit the database overflow state. If this parameter is 0, router exits the overflow state only after an explicit administrator command.

Mode Router Configuration

Usage Use this command to limit the number of AS-external-LSAs a router can receive, once it is in the wait state. It takes the number of seconds specified as the *<recover-time>* to recover from this state.

Example The following example shows setting the maximum number of LSAs to 5 and the time to recover from overflow state to be 3:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# overflow database external 50 3
```

passive-interface (OSPF)

Use this command to suppress the sending of Hello packets on all interfaces, or on a specified interface. If you use the **passive-interface** command without the optional parameters then **all** interfaces are put into passive mode.

Use the **no** variant of this command to allow the sending of Hello packets on all interfaces, or on the specified interface. If you use the **no** variant of this command without the optional parameters then **all** interfaces are removed from passive mode.

Syntax

```
passive-interface [<interface>][<ip-address>]
no passive-interface [<interface>][<ip-address>]
```

Parameter	Description
<interface>	The name of the interface.
<ip-address>	IP address of the interface, entered in the form A.B.C.D.

Mode Router Configuration

Usage Configure an interface to be passive if you wish its connected route to be treated as an OSPF route (rather than an AS-external route), but do not wish to actually exchange any OSPF packets via this interface.

Examples To configure passive interface mode on interface **vlan2**, enter the following commands:

```
awplus(config)# router ospf 100
awplus(config-router)# passive-interface vlan2
```

To configure passive interface mode on **all** interfaces, enter the following commands:

```
awplus(config)# router ospf 100
awplus(config-router)# passive-interface
```

To remove passive interface mode on interface **vlan2**, enter the following commands:

```
awplus(config)# router ospf 100
awplus(config-router)# no passive-interface vlan2
```

To remove passive interface mode on **all** interfaces, enter the following commands:

```
awplus(config)# router ospf 100
awplus(config-router)# no passive-interface
```

redistribute (OSPF)

Use this command to redistribute routes from other routing protocols, static routes and connected routes into an ospf routing table.

Use the **no** variant of this command to disable this function.

Syntax

```
redistribute {connected|rip|static} {metric|metric-type|route-map|
    tag}
no redistribute {connected|rip|static} {metric|metric-type|route-map|
    tag}
```

Parameter	Description
`connected`	Specifies that this applies to the redistribution of connected routes.
`rip`	Specifies that this applies to the redistribution of RIP routes.
`static`	Specifies that this applies to the redistribution of static routes.
`metric`	`metric <0-16777214>` Specifies the external metric.
`metric-type`	`metric-type {1|2}` Specifies the external metric-type.
`route-map`	`route-map WORD` Specifies name of the route-map.
`tag`	`tag <0-4294967295>` Specifies the external route tag.

Default The default metric value for routes redistributed into OSPF is 20. The metric can also be defined using the **set metric** command for a route map. Note that a metric defined using the **set metric** command for a route map overrides a metric defined with this command.

Mode Router Configuration

Usage You use this command to inject routes, learned from other routing protocols, into the OSPF domain to generate AS-external-LSAs. If a route-map is configured by this command, then that route-map is used to control which routes are redistributed and can set metric and tag values on particular routes.

The metric, metric-type, and tag values specified on this command are applied to any redistributed routes that are not explicitly given a different metric, metric-type, or tag value by the route map.

See the section **OSPF Metrics** in the **OSPF Introduction and Configuration** chapter for more information about metrics, and about behavior when configured in route maps.

Example The following example shows the configuration of a route-map named `rmap2`, which is then applied using the **redistribute route-map** command, so routes learned via interface vlan1 can be redistributed as type-1 external LSAs:

```
awplus# configure terminal
awplus(config)# route-map rmap2 permit 3
awplus(config-route-map)# match interface vlan1
awplus(config-route-map)# set metric-type 1
awplus(config-route-map)# exit
awplus(config)# router ospf 100
awplus(config-router)# redistribute rip route-map rmap2
```

Note that configuring a route-map and applying it with the **redistribute route-map** command allows you to filter which routes are distributed from another routing protocol (such as RIP). A route-map can also set the metric, tag, and metric-type of the redistributed routes.

Validation Commands **show ip ospf database external**

Related Commands **distribute-list (OSPF)**
match interface
route-map

restart ospf graceful

Use this command to force the OSPF process to restart, and optionally set the grace-period.

Syntax `restart ospf graceful [grace-period <1-1800>]`

Parameter	Description
`grace-period`	Specify the grace period.
<1-1800>	The grace period in seconds.

Default In the AlliedWare Plus™ OSPF implementation, the default OSPF grace-period is 180 seconds.

Mode Privileged Exec

Usage After this command is executed, the OSPF process immediately shuts down. It notifies the system that OSPF has performed a graceful shutdown. Routes installed by OSPF are preserved until the grace-period expires.

When a **restart ospf graceful** command is issued, the OSPF configuration is reloaded from the last saved configuration. Ensure you first enter the command **copy running-config startup-config**.

When a master failover happens on a VCStack, the OSPF grace-period will be the longest period between the default value (180 seconds is the default OSPF grace-period) and the configured OSPF grace-period value from this command. So the configured OSPF grace-period value will not be used for a VCStack master failover if it is shorter than the default OSPF grace-period.

Example

```
awplus# copy running-config startup-config
awplus# restart ospf graceful grace-period 200
```

Related Commands **ospf restart grace-period**
ospf restart helper

router ospf

Use this command to enter Router Configuration mode to configure an OSPF routing process. You must specify the process ID with this command for multiple OSPF routing processes on the switch.

Use the **no** variant of this command to terminate an OSPF routing process.

Use the **no** parameter with the **process-id** parameter, to terminate and delete a specific OSPF routing process. If no **process-id** is specified on the **no** variant of this command, then all OSPF routing processes are terminated, and all OSPF configuration is removed.

Syntax

```
router ospf [<process-id>]
no router ospf [<process-id>]
```

Parameter	Description
<process-id>	A positive number from 1 to 65535, that is used to define a routing process.

Default No routing process is defined by default.

Mode Global Configuration

Usage The process ID of OSPF is an optional parameter for the **no** variant of this command only. When removing all instances of OSPF, you do not need to specify each Process ID, but when removing particular instances of OSPF you must specify each Process ID to be removed.

Example To enter Router Configuration mode to configure an existing OSPF routing process `100`, use the commands:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)#
```

router-id

Use this command to specify a router ID for the OSPF process.

Use the **no** variant of this command to force OSPF to use the previous OSPF router-id behavior.

Syntax

```
router-id <ip-address>
no router-id
```

Parameter	Description
<ip-address>	Specifies the router ID in IPv4 address format.

Mode Router Configuration

Usage Configure each router with a unique router-id. In an OSPF router process that has active neighbors, a new router-id is used at the next reload or when you restart OSPF manually.

Example The following example shows a fixed router ID 10.10.10.60

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# router-id 10.10.10.60
```

Related Commands show ip ospf

show debugging ospf

Use this command to display which OSPF debugging options are currently enabled.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show debugging ospf
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show debugging ospf
```

Output

Figure 42-2: Example output from the show debugging ospf command

```
OSPF debugging status:
  OSPF packet Link State Update debugging is on
  OSPF all events debugging is on
```

show ip ospf

Use this command to display general information about all OSPF routing processes. Include the process ID parameter with this command to display information about specified instances.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf
show ip ospf <process-id>
```

Parameter	Description
<process-id>	*<0-65535>* The ID of the router process for which information will be displayed. If this parameter is included, only the information for the specified routing process is displayed.

Mode User Exec and Privileged Exec

Examples To display general information about all OSPF routing processes, use the command:

```
awplus# show ip ospf
```

To display general information about OSPF routing process `100`, use the command:

```
awplus# show ip ospf 100
```

Table 42-1: Example output from the show ip ospf command

```
Route Licence: Route : Limit=0, Allocated=0, Visible=0, Internal=0
Route Licence: Breach: Current=0, Watermark=0
Routing Process "ospf 10" with ID 192.168.1.1
Process uptime is 10 hours 24 minutes
Process bound to VRF default
Conforms to RFC2328, and RFC1583 Compatibility flag is disabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Graceful Restart
SPF schedule delay min 0.500 secs, SPF schedule delay max 50.0 secs
Refresh timer 10 secs
Number of incomming current DD exchange neighbors 0/5
Number of outgoing current DD exchange neighbors 0/5
Number of external LSA 0. Checksum 0x000000
Number of opaque AS LSA 0. Checksum 0x000000
Number of non-default external LSA 0
External LSA database is unlimited.
Number of LSA originated 0
Number of LSA received 0
Number of areas attached to this router: 2
   Area 0 (BACKBONE) (Inactive)
        Number of interfaces in this area is 0(0)
        Number of fully adjacent neighbors in this area is 0
        Area has no authentication
        SPF algorithm executed 0 times
        Number of LSA 0. Checksum 0x000000
```

Table 42-1: Example output from the show ip ospf command (cont.)

```
    Area 1 (Inactive)
        Number of interfaces in this area is 0(0)
        Number of fully adjacent neighbors in this area is 0
        Number of fully adjacent virtual neighbors through this area is 0
        Area has no authentication
        SPF algorithm executed 0 times
        Number of LSA 0. Checksum 0x000000
```

Table 42-2: Example output from the show ip ospf <process-id> command

```
Routing Process "ospf 100" with ID 10.10.11.146
Process uptime is 0 minute
Conforms to RFC2328, and RFC1583Compatibility flag is disabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Refresh timer 10 secs
Number of external LSA 0. Checksum Sum 0x0
Number of non-default external LSA 0
External LSA database is unlimited.
Number of areas attached to this router: 1
   Area 1
        Number of interfaces in this area is 1(1)
        Number of fully adjacent neighbors in this area is 0
        Number of fully adjacent virtual neighbors through this area is 0
        Area has no authentication
        SPF algorithm executed 0 times
        Number of LSA 1. Checksum Sum 0x00e3e2
```

Table 42-3: Parameters in the output of the show ip ospf command

Output Parameter		Meaning
`Route Licence: Route:`	`Limit`	The maximum number of OSPF routes which may be used for forwarding.
	`Allocated`	The current total number of OSPF routes allocated in the OSPF module.
	`Visible`	The current number of OSPF routes which may be used for forwarding.
	`Internal`	The number of OSPF internal routes used for calculating paths to ASBRs.
`Number of external LSA`		The number of external link-state advertisements
`Number of opaque AS LSA`		Number of opaque link-state advertisements

Related Commands router ospf

show ip ospf border-routers

Use this command to display the ABRs and ASBRs for all OSPF instances. Include the process ID parameter with this command to view data about specified instances.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf border-routers
```

```
show ip ospf <process-id> border-routers
```

Parameter	Description
<process-id>	*<0-65535>* The ID of the router process for which information will be displayed.

Mode User Exec and Privileged Exec

Output **Figure 42-3: Example output from the show ip ospf border-routers command**

```
OSPF process 1 internal Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 10.15.0.1 [10] via 10.10.0.1, vlan2, ASBR, Area 0.0.0.0
i 172.16.10.1 [10] via 10.10.11.50, vlan3, ABR, ASBR, Area
0.0.0.0
```

show ip ospf database

Use this command to display a database summary for OSPF information. Include the process ID parameter with this command to display information about specified instances.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf [<process-id>] database [self-originate|max-age|adv
router <adv-router-id>]
```

Parameter	Description
<process-id>	*<0-65535>* The ID of the router process for which information will be displayed.
self-originate	Displays self-originated link states.
max-age	Displays LSAs in MaxAge list. It maintains the list of the all LSAs in the database which have reached the max-age which is 3600 seconds.
adv-router	Advertising Router LSA.
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.

Mode User Exec and Privileged Exec

Examples To display the ABRs and ASBRs for all OSPF instances, use the command:

```
awplus# show ip ospf border-routers
```

To display the ABRs and ASBRs for the specific OSPF instance 721, use the command:

```
awplus# show ip ospf 721 border-routers
```

Output **Figure 42-4: Example output from the show ip ospf database command**

```
       OSPF Router process 1 with ID (10.10.11.60)
                Router Link States (Area 0.0.0.1)
Link ID         ADV Router      Age  Seq#       CkSum  Link
count
10.10.11.60     10.10.11.60       32 0x80000002 0x472b 1
       OSPF Router process 100 with ID (10.10.11.60)
                Router Link States (Area 0.0.0.0)
Link ID         ADV Router      Age  Seq#       CkSum  Link
count
10.10.11.60     10.10.11.60      219 0x80000001 0x4f5d 0
```

Example

```
awplus# show ip ospf database external 1.2.3.4 self-originate
awplus# show ip ospf database self-originate
```

Figure 42-5: Example output from the show ip ospf database self-originate command

```
       OSPF Router process 100 with ID (10.10.11.50)
                Router Link States (Area 0.0.0.1 [NSSA])
Link ID         ADV Router      Age  Seq#       CkSum  Link
count
10.10.11.50     10.10.11.50     20  0x80000007  0x65c3 2
                Area-Local Opaque-LSA (Area 0.0.0.1 [NSSA])
Link ID         ADV Router      Age  Seq#       CkSum Opaque ID
67.1.4.217      10.10.11.50     37 0x80000001  0x2129 66777
                AS-Global Opaque-LSA
Link ID         ADV Router      Age  Seq#       CkSum Opaque ID
67.1.4.217      10.10.11.50     37 0x80000001  0x2daa 66777
```

show ip ospf database asbr-summary

Use this command to display information about the Autonomous System Boundary Router (ASBR) summary LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database asbr-summary [<ip-addr>] [self-originate|
   <advrouter>]
```

Parameter	Description
<advrouter>	`adv-router` **<***ip-address***>**
`adv-router`	Displays all the LSAs of the specified router.
<ip-addr>	A link state ID, as an IP address.
`self-originate`	Displays self-originated link states.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database asbr-summary 1.2.3.4 self-
originate

awplus# show ip ospf database asbr-summary self-originate

awplus# show ip ospf database asbr-summary 1.2.3.4 adv-router
2.3.4.5
```

show ip ospf database external

Use this command to display information about the external LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database external adv-router[<adv-router-id>]
    [self-originate|adv-router<adv-router-id>]
```

Parameter	Description
`adv-router`	Displays all the LSAs of the specified router.
`self-originate`	Displays self-originated link states.
`<adv-router-id>`	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database external 1.2.3.4 self-originate

awplus# show ip ospf database external self-originate

awplus# show ip ospf database external 1.2.3.4 adv-router
        2.3.4.5
```

Output **Figure 42-6: Example output from the show ip ospf database external self-originate command**

```
       OSPF Router process 100 with ID (10.10.11.50)
                AS External Link States
  LS age: 298
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: AS-external-LSA
  Link State ID: 10.10.100.0 (External Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x7033
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        TOS: 0
        Metric: 20
        Forward Address: 10.10.11.50
        External Route Tag: 0
```

Output

Figure 42-7: Example output from the show ip ospf database external adv-router command

```
awplus#show ip ospf database external adv-router 1.1.1.1

                  AS External Link States
LS age: 273
Options: 0x2 (-|-|-|-|-|-|E|-)
LS Type: AS-external-LSA
Link State ID: 172.16.0.0 (External Network Number)
Advertising Router: 1.1.1.1
LS Seq Number: 80000004
Checksum: 0x02f8
Length: 36
Network Mask: /24
      Metric Type: 2 (Larger than any link state path)
      TOS: 0
      Metric: 20
      Forward Address: 0.0.0.0
      External Route Tag: 0
```

show ip ospf database network

Use this command to display information about the network LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database network [<adv-router-id>] [self-originate|
   <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The router ID of the advertising router, in IPv4 address format. Note however, that this no longer represents a real address.
`self-originate`	Displays self-originated link states.
`adv-router`	Displays all the LSAs of the specified router.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database network 1.2.3.4 self-originate
awplus# show ip ospf database network self-originate
awplus# show ip ospf database network 1.2.3.4 adv-router
       2.3.4.5
```

Output **Figure 42-8: Example output from the show ip ospf database network command**

```
      OSPF Router process 200 with ID (192.30.30.2)
               Net Link States (Area 0.0.0.0)
LS age: 1387
Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: network-LSA
Link State ID: 192.10.10.9 (address of Designated Router)
Advertising Router: 192.30.30.3
LS Seq Number: 80000001
Checksum: 0xe1b0
Length: 32
Network Mask: /24
      Attached Router: 192.20.20.1
      Attached Router: 192.30.30.3
LS age: 1648
Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: network-LSA
Link State ID: 192.30.30.3 (address of Designated Router)
Advertising Router: 192.30.30.3
LS Seq Number: 8000000f
Checksum: 0xe864
Length: 32
Network Mask: /24
      Attached Router: 192.30.30.2
      Attached Router: 192.30.30.3
```

Figure 42-9: Example output from the show ip ospf database network command

```
     OSPF Router process 200 with ID (192.30.30.2)
               Net Link States (Area 0.0.0.0)
LS age: 1175
Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: network-LSA
Link State ID: 192.10.10.9 (address of Designated Router)
Advertising Router: 192.30.30.3
LS Seq Number: 80000002
Checksum: 0xdfb1
Length: 32
Network Mask: /24
      Attached Router: 192.20.20.1
      Attached Router: 192.30.30.3
LS age: 1327
Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: network-LSA
Link State ID: 192.20.20.2 (address of Designated Router)
Advertising Router: 192.20.20.2
LS Seq Number: 8000000d
Checksum: 0xbce6
Length: 32
Network Mask: /24
      Attached Router: 192.20.20.1
      Attached Router: 192.20.20.2
LS age: 1278
Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: network-LSA
Link State ID: 192.30.30.3 (address of Designated Router)
Advertising Router: 192.30.30.3
Advertising Router: 192.30.30.3
LS Seq Number: 80000001
Checksum: 0x0556
Length: 32
Network Mask: /24
      Attached Router: 192.30.30.2
      Attached Router: 192.30.30.3
LS age: 1436
Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: network-LSA
Link State ID: 192.40.40.2 (address of Designated Router)
Advertising Router: 192.20.20.2
LS Seq Number: 8000000e
Checksum: 0xf173
Length: 32
Network Mask: /24
      Attached Router: 192.20.20.2
      Attached Router: 192.30.30.2
```

show ip ospf database nssa-external

Use this command to display information about the NSSA external LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database nssa-external [<ip-address>] [self-originate|
   <advrouter>]
```

Parameter	Description
<advrouter>	`adv-router` **<***ip-address***>**
`adv-router`	Displays all the LSAs of the specified router.
<ip-address>	A link state ID, as an IP address.
`self-originate`	Displays self-originated link states.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database nssa-external 1.2.3.4
       self-originate

awplus# show ip ospf database nssa-external self-originate

awplus# show ip ospf database nssa-external 1.2.3.4
       adv-router 2.3.4.5
```

Output

Figure 42-10: Example output from the show ip ospf database nssa-external adv-router command

```
      OSPF Router process 100 with ID (10.10.11.50)
               NSSA-external Link States (Area 0.0.0.0)
               NSSA-external Link States (Area 0.0.0.1 [NSSA])
 LS age: 78
 Options: 0x0 (*|-|-|-|-|-|-|-)
 LS Type: AS-NSSA-LSA
 Link State ID: 0.0.0.0 (External Network Number For NSSA)
 Advertising Router: 10.10.11.50
 LS Seq Number: 80000001
 Checksum: 0xc9b6
 Length: 36
 Network Mask: /0
       Metric Type: 2 (Larger than any link state path)
       TOS: 0
       Metric: 1
       NSSA: Forward Address: 0.0.0.0
--More--
      OSPF Router process 100 with ID (10.10.11.50)
               NSSA-external Link States (Area 0.0.0.0)
               NSSA-external Link States (Area 0.0.0.1 [NSSA])
 LS age: 78
 Options: 0x0 (*|-|-|-|-|-|-|-)
 LS Type: AS-NSSA-LSA
 Link State ID: 0.0.0.0 (External Network Number For NSSA)
 Advertising Router: 10.10.11.50
 LS Seq Number: 80000001
 Checksum: 0xc9b6
 Length: 36
 Network Mask: /0
       Metric Type: 2 (Larger than any link state path)
       TOS: 0
       Metric: 1
       NSSA: Forward Address: 0.0.0.0
       External Route Tag: 0
                NSSA-external Link States (Area 0.0.0.1 [NSSA])
```

show ip ospf database opaque-area

Use this command to display information about the area-local (link state type 10) scope LSAs. Type-10 Opaque LSAs are not flooded beyond the borders of their associated area.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database opaque-area [<ip-address>] [self-originate|
   <advrouter>]
```

Parameter	Description
<advrouter>	adv-router **<***ip-address***>**
adv-router	Displays all the LSAs of the specified router.
<ip-address>	A link state ID, as an IP address.
self-originate	Displays self-originated link states.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database opaque-area 1.2.3.4 self-
originate

awplus# show ip ospf database opaque-area self-originate

awplus# show ip ospf database opaque-area 1.2.3.4 adv-router
2.3.4.5
```

Output **Figure 42-11: Example output from the show ip ospf database opaque-area command**

```
      OSPF Router process 100 with ID (10.10.11.50)
                Area-Local Opaque-LSA (Area 0.0.0.0)
  LS age: 262
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: Area-Local Opaque-LSA
  Link State ID: 10.0.25.176 (Area-Local Opaque-Type/ID)
  Opaque Type: 10
  Opaque ID: 6576
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0xb413
  Length: 26
```

show ip ospf database opaque-as

Use this command to display information about the link-state type 11 LSAs. This type of link-state denotes that the LSA is flooded throughout the Autonomous System (AS).

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database opaque-as [<ip-address>] [self-originate|
    <advrouter>]
```

Parameter	Description
<advrouter>	adv-router **<***ip-address***>**
adv-router	Displays all the LSAs of the specified router.
<ip-address>	A link state ID, as an IP address.
self-originate	Displays self-originated link states.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database opaque-as 1.2.3.4 self-
        originate

awplus# show ip ospf database opaque-as self-originate

awplus# show ip ospf database opaque-as 1.2.3.4 adv-router
        2.3.4.5
```

Output **Figure 42-12: Example output from the show ip ospf database opaque-as command**

```
        OSPF Router process 100 with ID (10.10.11.50)
                AS-Global Opaque-LSA
  LS age: 325
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: AS-external Opaque-LSA
  Link State ID: 11.10.9.23 (AS-external Opaque-Type/ID)
  Opaque Type: 11
  Opaque ID: 657687
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0xb018
  Length: 25
```

show ip ospf database opaque-link

Use this command to display information about the link-state type 9 LSAs. This type denotes a link-local scope. The LSAs are not flooded beyond the local network.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database opaque-link [<ip-address>] [self-originate|
   <advrouter>]
```

Parameter	Description
<advrouter>	adv-router **<***ip-address***>**
adv-router	Displays all the LSAs of the specified router.
<ip-address>	A link state ID, as an IP address.
self-originate	Displays self-originated link states.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database opaque-link 1.2.3.4 self-
       originate

awplus# show ip ospf database opaque-link self-originate

awplus# show ip ospf database opaque-link 1.2.3.4 adv-router
       2.3.4.5
```

Output **Figure 42-13: Example output from the show ip ospf database opaque-link command**

```
       OSPF Router process 100 with ID (10.10.11.50)
                Link-Local Opaque-LSA (Link hme0:10.10.10.50)
  LS age: 276
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: Link-Local Opaque-LSA
  Link State ID: 10.0.220.247 (Link-Local Opaque-Type/ID)
  Opaque Type: 10
  Opaque ID: 56567
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x744e
  Length: 26
                Link-Local Opaque-LSA (Link hme1:10.10.11.50)
```

show ip ospf database router

Use this command to display information only about the router LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip ospf database router [<adv-router-id> self-originate|<adv-router-id>]`

Parameter	Description
`adv-router`	Displays all the LSAs of the specified router.
`self-originate`	Displays self-originated link states.
`<adv-router-id>`	The router ID of the advertising router, in IPv4 address format. Note however, that this no longer represents a real address.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database router 1.2.3.4 self-originate

awplus# show ip ospf database router self-originate

awplus# show ip ospf database router 1.2.3.4 adv-router
        2.3.4.5
```

Output **Figure 42-14: Example output from the show ip ospf database router command**

```
       OSPF Router process 100 with ID (10.10.11.50)
                Router Link States (Area 0.0.0.0)
LS age: 878
Options: 0x2 (*|-|-|-|-|-|E|-)
Flags: 0x3 : ABR ASBR
LS Type: router-LSA
Link State ID: 10.10.11.50
Advertising Router: 10.10.11.50
LS Seq Number: 80000004
Checksum: 0xe39e
Length: 36
 Number of Links: 1
  Link connected to: Stub Network
   (Link ID) Network/subnet number: 10.10.10.0
   (Link Data) Network Mask: 255.255.255.0
    Number of TOS metrics: 0
     TOS 0 Metric: 10
                Router Link States (Area 0.0.0.1)
LS age: 877
Options: 0x2 (*|-|-|-|-|-|E|-)
Flags: 0x3 : ABR ASBR
LS Type: router-LSA
Link State ID: 10.10.11.50
Advertising Router: 10.10.11.50
LS Seq Number: 80000003
Checksum: 0xee93
Length: 36
 Number of Links: 1
  Link connected to: Stub Network
   (Link ID) Network/subnet number: 10.10.11.0
   (Link Data) Network Mask: 255.255.255.0
    Number of TOS metrics: 0
     TOS 0 Metric: 10
```

show ip ospf database summary

Use this command to display information about the summary LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf database summary [<ip-address>] [self-originate|
   <advrouter>]
```

Parameter	Description
<advrouter>	adv-router **<***ip-address***>**
adv-router	Displays all the LSAs of the specified router.
<ip-address>	A link state ID, as an IP address.
self-originate	Displays self-originated link states.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf database summary 1.2.3.4 self-originate

awplus# show ip ospf database summary self-originate

awplus# show ip ospf database summary 1.2.3.4 adv-router
        2.3.4.5
```

Output **Figure 42-15: Example output from the show ip ospf database summary command**

```
      OSPF Router process 100 with ID (10.10.11.50)
                Summary Link States (Area 0.0.0.0)
                Summary Link States (Area 0.0.0.1)
  LS age: 1124
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: summary-LSA
  Link State ID: 10.10.10.0 (summary Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x41a2
  Length: 28
  Network Mask: /24
        TOS: 0  Metric: 10
```

Figure 42-16: Example output from the show ip ospf database summary self-originate command

```
       OSPF Router process 100 with ID (10.10.11.50)
                Summary Link States (Area 0.0.0.0)
  LS age: 1061
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: summary-LSA
  Link State ID: 10.10.11.0 (summary Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x36ac
  Length: 28
  Network Mask: /24
        TOS: 0  Metric: 10
                Summary Link States (Area 0.0.0.1)
  LS age: 1061
  Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: summary-LSA
  Link State ID: 10.10.11.0 (summary Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x36ac
  Length: 28
  Network Mask: /24
        TOS: 0  Metric: 10
                Summary Link States (Area 0.0.0.1)
  LS age: 1061
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: summary-LSA
  Link State ID: 10.10.10.0 (summary Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x41a2
  Length: 28
  Network Mask: /24
        TOS: 0  Metric: 10
```

Figure 42-17: Example output from the show ip ospf database summary adv-router <*ip-address*> command

```
       OSPF Router process 100 with ID (10.10.11.50)
                Summary Link States (Area 0.0.0.0)
  LS age: 989
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: summary-LSA
  Link State ID: 10.10.11.0 (summary Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x36ac
  Length: 28
  Network Mask: /24
        TOS: 0  Metric: 10
               Summary Link States (Area 0.0.0.1)
  LS age: 989
  Options: 0x2 (*|-|-|-|-|-|E|-)
LS Type: summary-LSA
  Link State ID: 10.10.11.0 (summary Network Number)
  Advertising Router: 10.10.11.50
  LS Seq Number: 80000001
  Checksum: 0x36ac
  Length: 28
  Network Mask: /24
        TOS: 0  Metric: 10
```

show ip ospf interface

Use this command to display interface information for OSPF.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip ospf interface [<interface-name>]`

Parameter	Description
<interface-name>	The VLAN name, for example vlan3.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf interface vlan2
```

Output **Figure 42-18: Example output from the show ip ospf interface command**

```
vlan2 is up, line protocol is up
  Internet Address 1.1.1.1/24, Area 0.0.0.0, MTU 1500
  Process ID 0, Router ID 33.33.33.33, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Waiting, Priority 1, TE Metric 0
  No designated router on this network
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:02
  Neighbor Count is 0, Adjacent neighbor count is 0
  Crypt Sequence Number is 1106347721
  Hello received 0 sent 1, DD received 0 sent 0
  LS-Req received 0 sent 0, LS-Upd received 0 sent 0
  LS-Ack received 0 sent 0, Discarded 0
```

show ip ospf neighbor

Use this command to display information on OSPF neighbors. Include the **ospf-id** parameter with this command to display information about specified instances.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip ospf [<ospf-id>] neighbor <neighbor-ip-addr> [detail]
show ip ospf [<ospf-id>] neighbor detail [all]
show ip ospf [<ospf-id>] neighbor [all]
show ip ospf [<ospf-id>] neighbor interface <ip-addr>
```

Parameter	Description
<ospf-id>	<0-65535> The ID of the router process for which information will be displayed.
<neighbor-ip-addr>	The Neighbor ID, entered as an IP address.
all	Include downstatus neighbor.
detail	Detail of all neighbors.
<ip-addr>	IP address of the interface.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip ospf neighbor detail
awplus# show ip ospf neighbor 1.2.3.4
awplus# show ip ospf neighbor interface 10.10.10.50 detail
       all
```

Output Note that before a device enters OSPF Graceful Restart it first informs its OSPF neighbors. In the **show** output, the * symbol beside the **Dead Time** parameter indicates that the switch has been notified of a neighbor entering the graceful restart state, as shown in **Figure 42-20**.

Figure 42-19: Example output from the show ip ospf neighbor command

```
OSPF process 1:
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.10.10.50       1   Full/DR         00:00:38    10.10.10.50     vlan1
OSPF process 100:
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.10.11.50       1   Full/Backup     00:00:31    10.10.11.50     vlan2
awplus#show ip ospf 1 neighbor
OSPF process 1:
Neighbor ID     Pri   State           Dead Time   Address         Interface
10.10.10.50       1   Full/DR         00:00:38    10.10.10.50     vlan1
```

Figure 42-20: Example output from the show ip ospf <ospf-id> neighbor command

```
OSPF process 100:
Neighbor ID     Pri   State           Dead Time   Address         Interface
192.168.0.3      50   2-Way/DROther   00:01:59*   192.168.200.3   vlan200
```

Figure 42-21: Example output from the show ip ospf neighbor detail command

```
Neighbor 10.10.10.50, interface address 10.10.10.50
   In the area 0.0.0.0 via interface vlan5
   Neighbor priority is 1, State is Full, 5 state changes
   DR is 10.10.10.50, BDR is 10.10.10.10
   Options is 0x42 (*|O|-|-|-|-|E|-)
   Dead timer due in 00:00:38
   Neighbor is up for 00:53:07
   Database Summary List 0
   Link State Request List 0
   Link State Retransmission List 0
   Crypt Sequence Number is 0
   Thread Inactivity Timer on
   Thread Database Description Retransmission off
   Thread Link State Request Retransmission off
   Thread Link State Update Retransmission on
Neighbor 10.10.11.50, interface address 10.10.11.50
   In the area 0.0.0.0 via interface vlan2
   Neighbor priority is 1, State is Full, 5 state changes
   DR is 10.10.11.10, BDR is 10.10.11.50
   Options is 0x42 (*|O|-|-|-|-|E|-)
   Dead timer due in 00:00:31
   Neighbor is up for 00:26:50
   Database Summary List 0
   Link State Request List 0
   Link State Retransmission List 0
   Crypt Sequence Number is 0
   Thread Inactivity Timer on
   Thread Database Description Retransmission off
   Thread Link State Request Retransmission off
   Thread Link State Update Retransmission on
```

show ip ospf route

Use this command to display the OSPF routing table. Include the `process ID` parameter with this command to display the OSPF routing table for specified instances.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip ospf [<ospf-id>] route`

Parameter	Description
<ospf-id>	<0-65535> The ID of the router process for which information will be displayed. If this parameter is included, only the information for this specified routing process is displayed.

Mode User Exec and Privileged Exec

Examples To display the OSPF routing table, use the command:

```
awplus# show ip ospf route
```

Output **Figure 42-22: Example output from the show ip ospf route command for a specific process**

```
OSPF process 1:
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
O  10.10.0.0/24 [10] is directly connected, vlan1, Area 0.0.0.0
O  10.10.11.0/24 [10] is directly connected, vlan2, Area 0.0.0.0
O  10.10.11.100/32 [10] is directly connected, lo, Area 0.0.0.0
E2 10.15.0.0/24 [10/50] via 10.10.0.1, vlan1
IA 172.16.10.0/24 [30] via 10.10.11.50, vlan2, Area 0.0.0.0
E2 192.168.0.0/16 [10/20] via 10.10.11.50, vlan2
```

show ip ospf virtual-links

Use this command to display virtual link information.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip ospf virtual-links`

Mode User Exec and Privileged Exec

Examples To display virtual link information, use the command:

```
awplus# show ip ospf virtual-links
```

Output **Figure 42-23: Example output from the show ip ospf virtual-links command**

```
Virtual Link VLINK0 to router 10.10.0.9 is up
  Transit area 0.0.0.1 via interface vlan5
  Transmit Delay is 1 sec, State Point-To-Point,
  Timer intervals configured, Hello 10, Dead 40, Wait 40,
Retransmit 5
    Hello due in 00:00:02
    Adjacency state Full
Virtual Link VLINK1 to router 10.10.0.123 is down
  Transit area 0.0.0.1 via interface *
  Transmit Delay is 1 sec, State Down,
  Timer intervals configured, Hello 10, Dead 40, Wait 40,
Retransmit 5
    Hello due in inactive
    Adjacency state Down
```

show ip protocols ospf

Use this command to display OSPF process parameters and statistics.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip protocols ospf`

Mode User Exec and Privileged Exec

Examples To display OSPF process parameters and statistics, use the command:

```
awplus# show ip protocols ospf
```

Output **Figure 42-24: Example output from the show ip protocols ospf command**

```
Routing Protocol is "ospf 200"
  Invalid after 0 seconds, hold down 0, flushed after 0
  Outgoing update filter list for all interfaces is
    Redistributed kernel filtered by filter1
  Incoming update filter list for all interfaces is
  Redistributing: kernel
  Routing for Networks:
    192.30.30.0/24
    192.40.40.0/24
  Routing Information Sources:
    Gateway         Distance      Last Update
  Distance: (default is 110)
    Address         Mask            Distance List
```

summary-address

Use this command to summarize, or possibly suppress, external routes that have the specified address range.

Use the **no** variant of this command to stop summarizing, or suppressing, external routes that have the specified address range.

Syntax

```
summary-address <ip-addr/prefix-length> [not-advertise]
    [tag <0-4294967295>]

no summary-address <ip-addr/prefix-length> [not-advertise]
    [tag <0-4294967295>]
```

Parameter	Description
<ip-addr/prefix-length>	Specifies the base IP address of the summary address.The range of addresses given as IPv4 starting address and a prefix length.
`not-advertise`	Set the **not-advertise** option if you do not want OSPF to advertise either the summary address or the individual networks within the range of the summary address.
`tag <0-4294967295>`	The tag parameter specifies the tag value that OSPF places in the AS external LSAs created as a result of redistributing the summary route. The tag overrides tags set by the original route.

Default The default tag value for a summary address is 0.

Mode Router Configuration

Usage An address range is a pairing of an address and a mask that is almost the same as IP network number. For example, if the specified address range is 192.168.0.0/255.255.240.0, it matches: 192.168.1.0/24, 192.168.4.0/22, 192.168.8.128/25 and so on.

Redistributing routes from other protocols into OSPF requires the router to advertise each route individually in an external LSA. Use the `summary address` command to advertise one summary route for all redistributed routes covered by a specified network address and mask. This helps decrease the size of the OSPF link state database.

Ensure OSPF routes exist in the summary address range for advertisement before using this command.

Example The following example uses the `summary-address` command to aggregate external LSAs that match the network 172.16.0.0/16 and assign a Tag value of 3.

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# summary-address 172.16.0.0/16 tag 3
```

timers spf exp

Use this command to adjust route calculation timers using exponential back-off delays.

Use **no** form of this command to return to the default exponential back-off timer values.

Syntax

```
timers spf exp <min-holdtime> <max-holdtime>
no timers spf exp
```

Parameter	Description
<min-holdtime>	*<0-2147483647>* Specifies the minimum delay between receiving a change to the SPF calculation in milliseconds. The default SPF min-holdtime value is 50 milliseconds.
<max-holdtime>	*<0-2147483647>* Specifies the maximum delay between receiving a change to the SPF calculation in milliseconds. The default SPF max-holdtime value is 50 seconds.

Mode Router Configuration

Default The default SPF min-holdtime is 50 milliseconds. The default SPF max-holdtime is 40 seconds.

Usage This command configures the minimum and maximum delay time between the receipt of a topology change and the calculation of the Shortest Path First (SPF).

Examples To set the minimum delay time to 5 milliseconds and maximum delay time to 10 milliseconds, use the commands:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# timers spf exp 5 10
```

To reset the minimum and maximum delay times to the default values, use the commands:

```
awplus# configure terminal
awplus(config)# router ospf 100
awplus(config-router)# no timers spf exp
```

Related Commands timers spf exp

undebug ospf events

This command applies the functionality of the **no debug ospf events** command on page 42.19.

undebug ospf ifsm

This command applies the functionality of the **no debug ospf ifsm** command on page 42.20.

undebug ospf lsa

This command applies the functionality of the **no debug ospf lsa** command on page 42.21.

undebug ospf nfsm

This command applies the functionality of the **no debug ospf nfsm** command on page 42.22.

undebug ospf nsm

This command applies the functionality of the **no debug ospf nsm** command on page 42.23.

undebug ospf packet

This command applies the functionality of the **no debug ospf packet** command on page 42.24.

undebug ospf route

This command applies the functionality of the **no debug ospf route** command on page 42.25.

Chapter 43: OSPFv3 for IPv6 Introduction and Configuration

OSPFv3 Introduction 43.2
Features 43.2
Licensing 43.3
Routing Overview 43.3
OSPF Components 43.4
- Autonomous Systems 43.4
- Routing Areas 43.4
- Relationships Between Routers 43.4
- OSPFv3 Packet Types 43.4
- Link State Advertisements (LSAs) 43.8
- LSA Header 43.9
- OSPFv3 States 43.10
- OSPFv3 Metrics 43.10
- Automatic Cost Calculation 43.11
- Network Types 43.11
- Passive Interfaces 43.12
- Redistributing External Routes 43.12

Differences Between OSPFv2 and OSPFv3 ..... 43.13
- Protocol Processing Applied Per-Link Rather than Per-Subnet 43.13
- Removed Address Semantics 43.13
- Neighbors are Identified by Router ID 43.13
- New Link-Local Flooding Scope for Link State Advertisements 43.14
- Uses Link-Local Unicast Addresses 43.14
- Support Provided for Multiple OSPF Instances per Link 43.14
- Unknown LSAs are Handled More Effectively 43.14

OSPFv3 Authentication and Encryption Overview 43.15
- OSPFv3 Authentication and Encryption Support 43.16
- OSPFv3 Virtual Links 43.17

Configuring OSPFv3 ..... 43.18
- Example 1: Single-Area Network Configuration 43.19
- Example 2: Two-Area Network Configuration 43.26
- Setting Up the Metrics 43.28

Configuring OSPFv3 Authentication and Encryption 43.33
- Configuring OSPFv3 Authentication on a VLAN 43.34
- Configuring OSPFv3 Encryption on a VLAN 43.35
- Configuring OSPFv3 Authentication in an OSPFv3 Area 43.36
- Configuring OSPFv3 Encryption in an OSPFv3 Area 43.37
- Configuring OSPFv3 Authentication and Encryption for a Virtual Link 43.38
- OSPFv3 Authentication in an OSPFv3 Area 43.40
- OSPFv3 Encryption in an OSPFv3 Area 43.42
- OSPFv3 Authentication on a VLAN 43.44
- OSPFv3 Encryption on a VLAN 43.46
- OSPFv3 Authentication with Two VLANs 43.48
- OSPFv3 Encryption with Two VLANs 43.50
- OSPFv3 Authentication for a Virtual Link 43.52
- OSPFv3 Encryption for a Virtual Link 43.54

OSPFv3 Introduction

This chapter introduces OSPFv3 followed by basic configuration examples. OSPFv3 is documented in RFC 2740 and is in essence OSPFv2 with changes and additions necessary for OSPF's operation within an IPv6 network. This introduction focuses on the differences between the two protocol versions, rather than to re-introduce OSPF as a routing method. See the chapter, **Chapter 41, OSPF Introduction and Configuration** for an introductory overview of OSPF on AlliedWare Plus™ switches.

To see details on the OSPF commands used in the examples, refer to the chapter, **Chapter 44, OSPFv3 for IPv6 Commands**.

Features

Not version-specific

Open Shortest Path First (OSPF) is an Interior Gateway Routing Protocol, based on Shortest Path First (SPF) or link-state technology. OSPF is defined in RFCs 1245–1247, 1253 and 1583. The protocol was designed specifically for the TCP/IP Internet environment, and supports the following features:

- Authentication of routing updates - authentication for OSPFv3 is achieved using IPsec AH and IPsec ESP header capabilities available in the IPv6 protocol.
- Tagging of externally-derived routes.
- Fast response to topology changes with low overhead.
- Load sharing over meshed links.

Specific to OSPFv3

The specific features and enhancements of OSPFv3 are introduced to enable OSPF to run over IPv6 networks. These aspects are covered later in this chapter.

Licensing

OSPF is a licensed feature, and can be obtained either with a "**Base License**" or a "**Full License.**" The table below shows the basic licensing structure for running OSPFv3. Note that licensing structures can vary with sales regions. We therefore recommend that you discuss your licensing options with your local Allied Telesis distributor or reseller.

OSPF Version	Base License	Full License
OSPFv3	Not bundled as part of the AlliedWare Plus release, but is separately available as a basic license pack that will support up to 64 routes.	Purchased separately. There is no licensed route limit, but route numbers are limited by switch specifications. See your switch's Datasheet for details.

Routing Overview

The basic function of a routing protocol is to establish the best path for packets to travel in order to reach their destination. The method of establishing the best path can be based on various different network measurements, such as the hop-count, bandwidth, or delay that a particular path may present, or could be based on static routes that are manually entered. This information is then fed into a mathematical algorithm such as those invented by Belman-Ford (for RIP), or Dijkstra (for OSPF), in order to calculate a best path to the destination.

A very brief overview of OSPF's general principles and components is presented in this chapter. For a further explanation refer to **Chapter 41, OSPF Introduction and Configuration**.

OSPF Components

Autonomous Systems

Routers combine to form Autonomous Systems (AS) that share a common management and protocol suite. Each router within an area has an identical database, which contains components that describe other routers, their paths and states.

Routing Areas

OSPF allows the grouping of networks into a set, called an **Area**. The internal topology of an area is hidden from the rest of the AS. This technique minimizes the routing traffic required for the protocol. When multiple areas are used, each area has its own copy of the topological database. Routing can be between areas (inter-area routing) or within areas (intra-area routing). A backbone area forms the central path between all other areas. A router that forms a connection between an area and the backbone is known as an area border router.

Relationships Between Routers

Neighbors and Adjacencies

By transmitting hello messages, routers that share a common physical connection initially form neighbor relationships. Neighbors that meet a common set of configuration factors can then form adjacent relationships through which they can exchange topological information.

Designated Router

On broadcast multi-access networks, one of the routers becomes a **designated router**. By issuing its link state advertisements, the designated router maintains adjacencies with all the routers within a broadcast domain. The designated router therefore coordinates and synchronizes the topological databases for all routers on the link.

The designated router is selected through the exchange of hello packets. The router transmitting the highest priority number will be elected the designated router. Where a tie exists, selection will be based on the highest transmitted router ID.

Backup Designated Router

For network efficiency and resiliency, the routers also elect a backup designated router, which is able to act as the designated router in the event of failure of the designated router.

OSPFv3 Packet Types

The OSPF protocol contains five basic packet types for its messages. The OSPFv3 protocol retains these basic types, although the structure of some of them has changed slightly. The basic message packet types are:

- hello packet
- database description packet
- link state request packet
- link state update packet
- link state acknowledgment packet

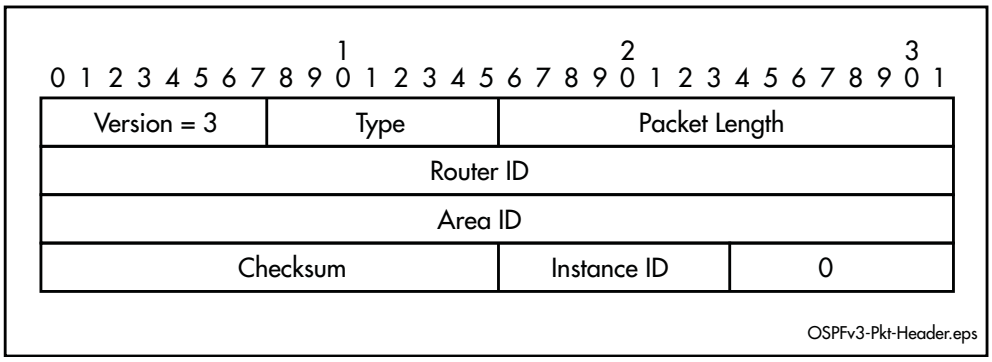
Also note that at the IP packet level, protocol number 89, previously assigned to OSPFv2 is now shared by both OSPFv2 and OSPFv3. These are the values contained within the protocol field of the IPv4 packets, and the Next Header field of IPv6 packets.

The five OSPFv3 message types are now described in greater detail.

OSPFv3 packet header

All OSPFv3 packets carry the common header component shown in **Figure 43-1**.

Figure 43-1: OSPFv3 common packet header



Because this packet component forms a common prefix for all the OSPFv3 packets, its components are expanded in greater detail than for the specific packet types that follow. **Table 43-1** shows the meaning of each field within the common header component.

Table 43-1: OSPFv3 common packet header components

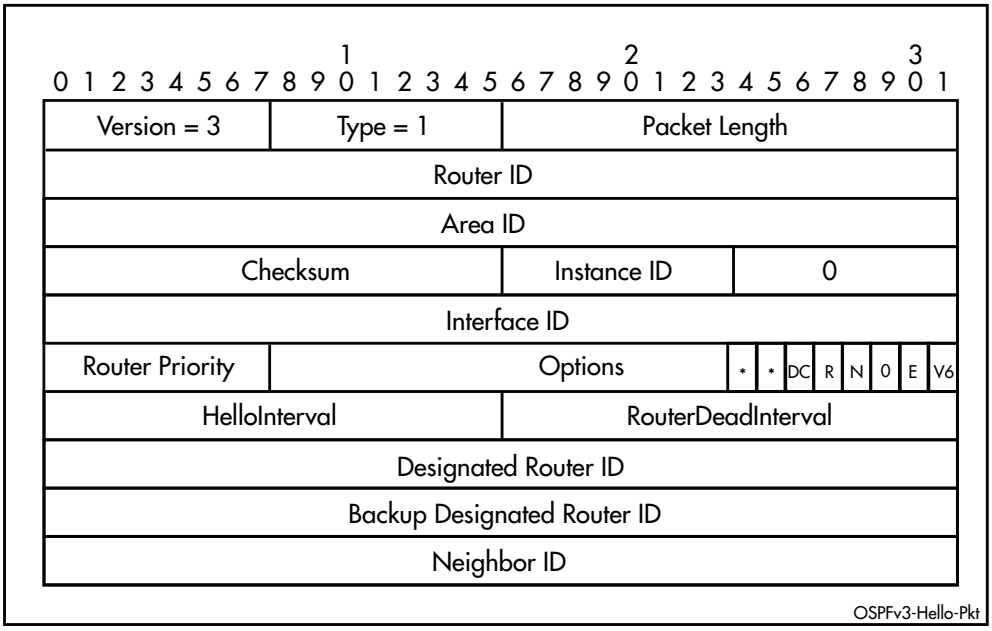
Packet Field	Purpose
Version	Denotes the OSPF version, i.e. 2 for OSPFv2, and 3 for OSPFv3.
Type	Denotes the packet type: « 1 = Hello packet « 2 = Database Description packet « 3 = Link State Request « 4 = Link State Update « 5 = Link State Acknowledgment
Packet Length	The packet length, in bytes, including the standard OSPFv3 header.
Router ID	The ID of the transmitting router.
Area ID	A 32 bit number that identifies the (single) area to which the packet belongs.
Checksum	The standard IPv6 packet checksum.
Instance ID	OSPFv3 is capable of running multiple OSPF instances over a single link. Each Instance is identified by its Instance ID. Note that the Instance ID is only of local (link) significance. Also note that the term “instance” is sometimes used in a totally different context in referring to specific LSA element components.
0	Reserved and normally set to 0.

Hello message packet

These messages are sent on all links. Their function is to establish and maintain neighbor relationships between routers.

Figure 43-2 shows the structure and content of the Hello packet. It follows the same basic format as the OSPFv2 hello packet, but without the Authentication or Network Mask fields, and with an additional field for the Instance ID.

Figure 43-2: OSPFv3 hello message type

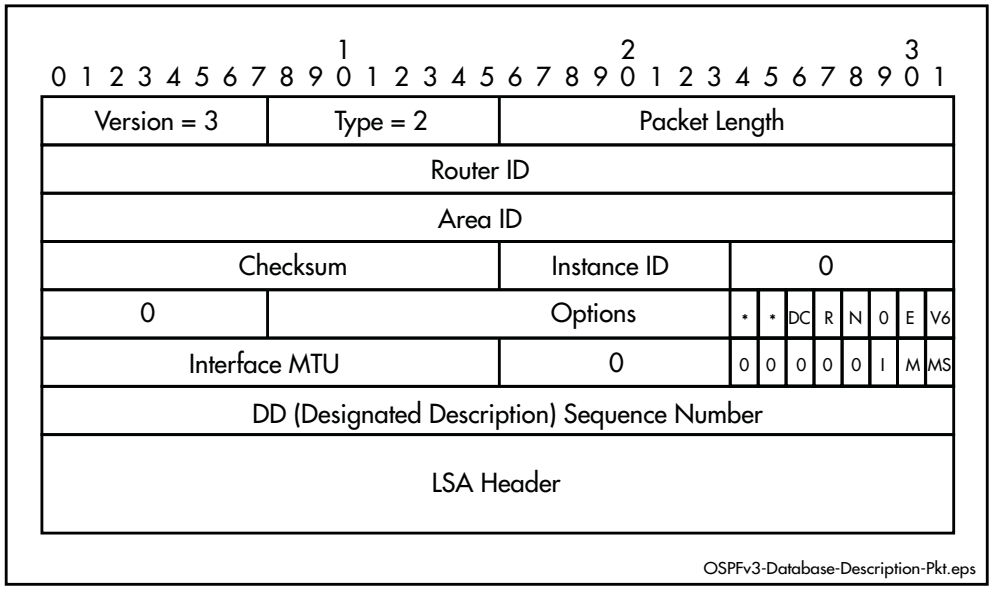


Database description packet

These packet types are exchanged as router adjacencies are established. They contain descriptions that define the link-state.

Figure 43-3 shows the structure and content of the database description packet. It follows the same basic format as that used for OSPFv2, but without the Authentication or Network Mask fields, and with an additional field for the Instance ID.

Figure 43-3: OSPFv3 database description packet

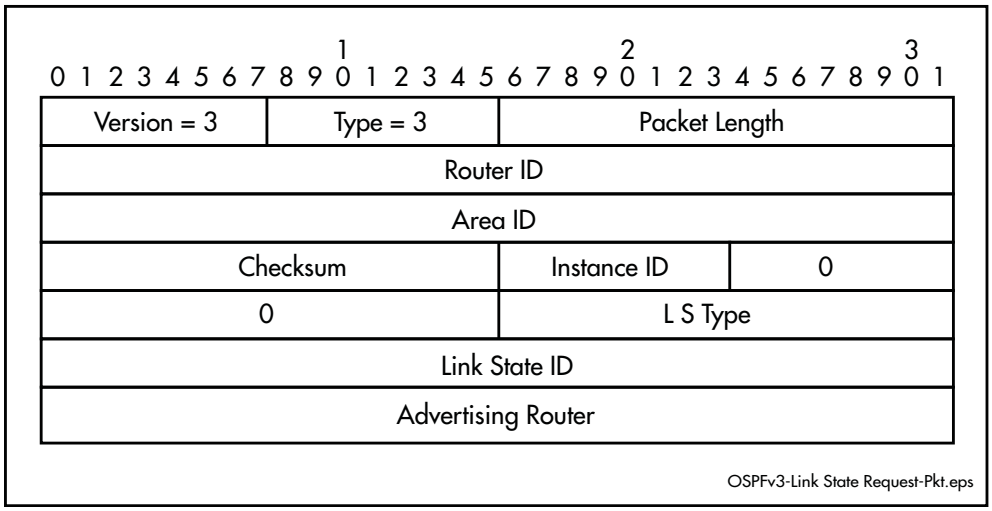


Link state request packet

Routers use these packets to request transmissions of specific portions (instances) of the Link State database, usually to replace out-of-date components. The requesting router knows exactly which database components it requires, based on their link state ID, sequence number, age, and checksum.

Figure 43-4 shows the structure and content of the a link state request packet. It follows the same basic format as that used for OSPFv2, but without the AuType, Authentication and network-mask fields, and with an additional field for the Instance ID.

Figure 43-4: OSPFv3 link-state request packet

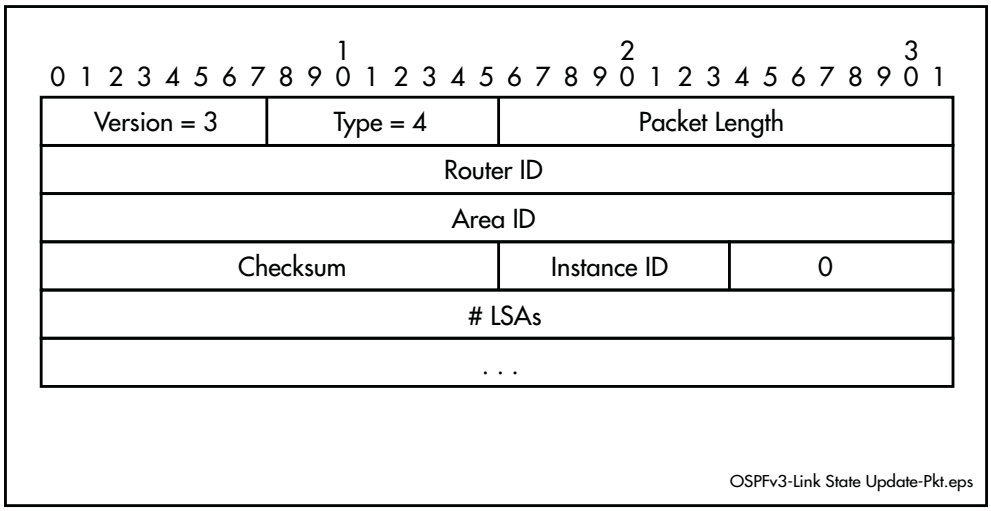


Link state update packet

These packet types are used to implement LSA flooding. They carry LSA information between a router and its immediate neighbors, often in response to link state requests.

Figure 43-5 shows the structure and content of the link state update packet. It follows the same basic format as that used for OSPFv2, but without the Authentication or Network-Mask fields, and with an additional field for the Instance ID.

Figure 43-5: OSPFv3 link-state update packet

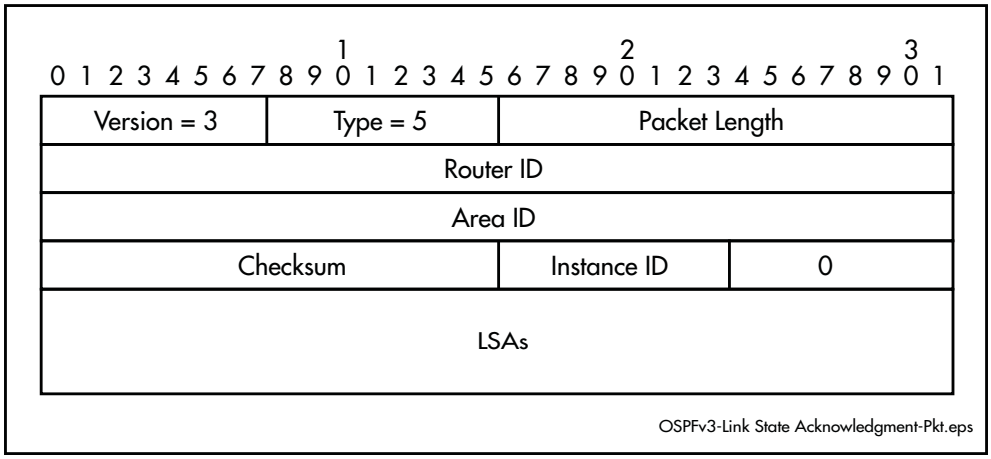


Link state acknowledgment packet

These multicast packet types are used to provide explicit acknowledgment of flooded LSAs. If a router transmits a link-state update, but does not receive acknowledgments from all neighbors, it will continue to retransmit the link state update until all neighbors have acknowledged it. This constitutes the guaranteed delivery mechanism within OSPF. A single packet can acknowledge multiple LSAs.

Figure 43-6 shows the structure and content of the link state acknowledgment packet. It follows the same basic format as that used for OSPFv2, but without the Authentication or Network Mask fields, and with an additional field for the Instance ID.

Figure 43-6: OSPFv3 link state acknowledgment packet



Link State Advertisements (LSAs)

Link state advertisements (LSAs) are records in the topological database. Nine different types of link state advertisements exist, see **Table 43-2**. Only the common header components are documented. A description of the LSAs themselves is outside the scope of this document.

Link state advertisements age to a maximum (MaxAge) of 3600 seconds, which limits their storage period in the topological database. When a link state advertisement reaches MaxAge, the router tries to flush it from the routing domain by reflooding the advertisement. A link state advertisement that has reached MaxAge is not used in further routing table calculations.

LSA types

Table 43-2 shows the LSA types supported by OSPFv3. Each type of link state advertisement describes a different set of features of the Autonomous System (AS).

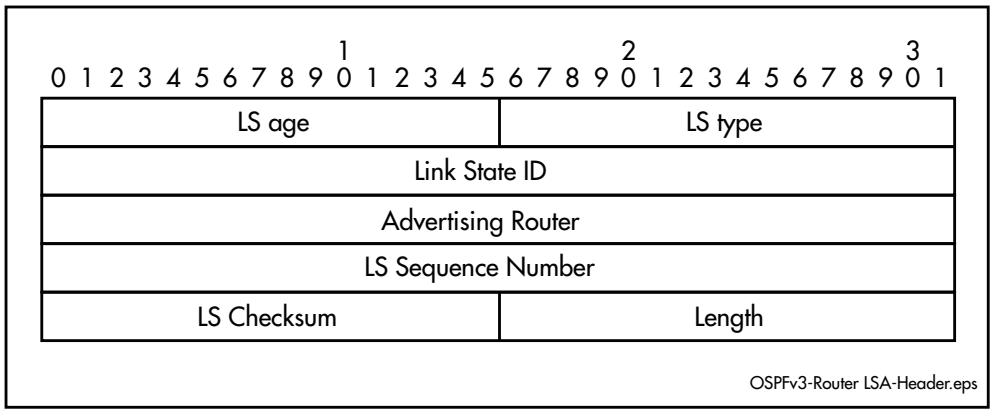
Table 43-2: OSPFv3 LSA header types

LSA Function Code	LS Type	Description
1	0x2001	Router LSA
2	0x2002	Network LSA
3	0x2003	Inter-Area-Prefix-LSA
4	0x2004	Inter-Area-Router-LSA
5	0x4005	AS-External-LSA
6	0x2006	Deprecated (may be reassigned)
7	0x2007	NSSA-LSA
8	0x0008	Link-LSA
9	0x2009	Intra-Area-Prefix-LSA
11	0x000b	Grace-LSA

LSA Header

Figure 43-7 shows the structure and content of the 20 byte header that is common to all OSPFv3 Link State Advertisements (LSAs). This header provides sufficient information to identify the link state type, its ID, and its advertising router.

Figure 43-7: OSPFv3 LSA header



OSPFv3 States

The following table describes the eight states that routers can pass through as they establish their neighbor relationships.

Table 43-3: OSPF states

Packet Type	Purpose
Down	This is the initial state. No hello packets have been received from the neighbor recently, or at all.
Attempt	This state applies to non-broadcast multi-access networks. The router is making a determined attempt to contact a statically configured neighbor. Hello packets are sent every hello interval.
Init	A hello packet has been seen from the neighbor, however the hello packet does not list the router as known.
2-Way	This state is entered when the communication between two neighbors is bidirectional (the hello packet from the neighbor lists this router as a neighbor).
ExStart	This is the first step in creating an adjacency between two routers. The two routers decide which is going to control the exchange between them.
Exchange	In this state, the neighbors exchange database description packets. Each packet summarizes the link state advertisements held by that router.
Loading	After all the database description information has been exchanged, the routers exchange link state advertisements required to update or complete each router's topological database, thereby synchronizing the two routers' databases.
Full	This is the final state, and the adjacency is complete. Reaching this state in itself may cause new instances of some link state advertisements, such as the network and router advertisements related to the two routers.

OSPFv3 Metrics

The metrics used by OSPF are not simple distance metrics, such as used by RIP for example, but are measurements of the path bandwidth. Interface metrics are normally set using the formula 10^8 / interface speed (in bps). This gives metrics such as 1 for a 100 Mbps Ethernet interface, and 1562 for a 64 kbps serial line.

When no metric is defined, the default metric for routes redistributed into OSPF is 20. You can define the metric using the **redistribute (IPv6 OSPF)** command.

You can also define the metric using the **set metric** command for a route map. If a route map is configured, but no OSPF metric is defined using a **redistribute (IPv6 OSPF)** command, then a default metric of 20 is used. If a route map is configured and the metric is defined using the **set metric** command, this supersedes a metric defined using the **redistribute (IPv6 OSPF)** command.

How to set the OSPFv3 metric when redistributing routes

You can determine the OSPFv3 metric using the following methods:

- use the default OSPFv3 metric, which is 20.
- change it using the **metric** parameter of the **redistribute (IPv6 OSPF)** command

The following example sets the metric to `5` for `connected` redistributed routes:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# redistribute connected metric 5
```

- change it using the **metric-value** parameter of the **set metric** command for a route map

Note that changing the metric using a route map supersedes the metric defined using the **redistribute (IPv6 OSPF)** command. The following example for entry `3` of the route map called `rmap1` give matching routes a metric of `5`:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# set metric 5
```

Automatic Cost Calculation

OSPF interfaces can automatically set the OSPF metric of an IP interface based on its bandwidth, instead of the system administrator having to manually set the OSPF metric. Automatic setting takes into account that the speed of an interface can change over time, when ports change link state or change speed via auto-negotiation or manual setting. If metrics are manually set, some interfaces are preferred when they should be changing to match dynamically changing network configurations.

Network Types

OSPF treats the networks attached to OSPF interfaces as one of the following network types, depending on the physical media:

- broadcast
- non-broadcast multi-access (NBMA)
- point-to-point
- point-to-multipoint
- virtual

By default, Ethernet networks are treated as broadcast networks. You can use the **ip ospf network** command on page 42.44 to configure a VLAN interface to be a different network type. Configure a VLAN or Ethernet interface as an Non-Broadcast Multi-Access (NBMA) interface when:

- some devices on the network do not support multicast addressing.
- you want to select which devices on the network are to become OSPF neighbors, rather than allow all the devices on the network to become OSPF neighbors.

Passive Interfaces

A passive interface does not take part in normal OSPF interface operations:

- OSPF does not transmit or receive Hello messages via the interface.
- The interface does not experience interface state transitions.
- OSPF does not associate neighbors with the interface.

If the interface is up, OSPF adds the network that is attached to the interface, as a stub network to the router LSA of the area in which the interface resides.

Examples To configure the passive interface mode on interface vlan2, enter the following commands:

```
awplus(config)# router ipv6 ospf
awplus(config-router)# passive-interface vlan2
```

To configure the passive interface mode on **all** interfaces, enter the following commands:

```
awplus(config)# router ipv6 ospf
awplus(config-router)# passive-interface
```

Redistributing External Routes

OSPFv3 can import and redistribute RIP, non-OSPF interface, and statically configured routes. It can also optionally assign the following settings to all routes it imports:

- a route metric
- the External metric type

Alternatively, you can assign a route map to select particular routes and set their route parameters. A route map can also filter out a subset of routes, so you do not have to import all routes.

The import settings also allow you to select whether to redistribute subnets (classless network routes), or only classful network routes.

To import and redistribute external routes into OSPF, create a route redistribution definition for the source routing protocol, using the **redistribute (IPv6 OSPF)** command on page 44.49.

Differences Between OSPFv2 and OSPFv3

Whilst the fundamental principles of OSPFv2 are maintained from those standardized for OSPF with IPv6 (OSPFv3), the following items, listed below, represent the major areas that are applied differently for OSPFv3. For more detail refer to RFC 2740 (OSPF for IPv6).

- Protocol processing applied per-link rather than per-subnet
- Removed address semantics
- Neighbors are identified by Router ID
- New link-local flooding scope for link state advertisements
- Uses link-local addresses
- Support provided for multiple instances per link
- OSPF-specific authentication has been removed
- Unknown LSAs are handled more effectively
- Authentication relies on IPsec features within the IPv6 protocol, rather than integrate authorization within the OSPF protocol itself (as is done for OSPFv2).
- Modified packet types
- Modified LSA types

The following sections expand on these points.

Protocol Processing Applied Per-Link Rather than Per-Subnet

OSPFv3 nodes use links to communicate at the link layer. You can also assign multiple subnets, and multiple IPv6 addresses, to a single link. In addition, two nodes can be assigned to and communicate over the same link without having to share the same subnet.

Thus the term **link** can be compared with the OSPFv2 terms **network** and **subnet.** This change has resulted in changes within the Hello packets and Network Link State Advertisements (LSAs).

Removed Address Semantics

Address semantics have been removed from OSPFv3 protocol packets and network Link State Advertisements (LSAs).

The intention of this change is to enable the core features to be protocol independent. The major changes in this respect are described below:

Note Although the 32 bit Router and ID fields still exist in IPv6 packets, and can still be entered in dotted decimal format, for OSPFv3, they are simply 32-bit-long labels, which no longer represent IP addresses.

Neighbors are Identified by Router ID

OSPFv2 neighbors were identified by IPv4 addresses on multipoint networks and by router ID on point-to-point networks. OSPFv3 removes this inconsistency by identifying all neighbors by their router ID.

New Link-Local Flooding Scope for Link State Advertisements

OSPFv3 contains the following flooding scopes:

- Area scope flooding (in OSPFv2)
- Autonomous System (AS) flooding (in OSPFv2)
- Link-local flooding (new in OSPFv3)

Uses Link-Local Unicast Addresses

On all OSPF interfaces except virtual links, OSPF packets are sent using the interface's associated link-local unicast address as the source address.

Support Provided for Multiple OSPF Instances per Link

An Instance field has been added to OSPFv3 hello and LSA packets. By assigning separate **instances**, multiple routers to be attached to a single broadcast link in such a way that separate adjacencies may be configured for selected router groups. See **"Configuring instances and processes" on page 43.31**.

Unknown LSAs are Handled More Effectively

In OSPFv2 routers discarded unknown LSA types. In OSPFv3 unknown LSA types are given either link-local flooding scope, or are stored and flooded as if being understood. The options determining which action to take, and flooding scope, is contained in components within the LSA type field.

OSPFv3 Authentication and Encryption Overview

To ensure that OSPFv3 packets are not modified, and are not being spoofed, configure OSPFv3 Authentication and Encryption on your devices, if they are running Version 5.4.3 or later AlliedWare Plus software.

OSPFv2 defines the AuType and Authentication fields in its protocol header to provide security. In OSPFv3, these OSPFv2 authentication fields were removed from OSPF headers. OSPFv3 requires the IPv6 AH (Authentication Header) header to provide authentication of routing information exchanges, and OSPFv3 requires the IPv6 ESP (Encapsulating Security Payload) header to provide encryption of routing information exchanges.

OSPFv3 Authentication and Encryption is specified in RFC 4552 “Authentication/ Confidentiality for OSPFv3”. RFC 4552 describes the AH/ESP extension headers.

See the section **Configuring OSPFv3 Authentication and Encryption** for OSPFv3 Authentication and Encryption configurations and topologies.

Also refer to **Chapter 44, OSPFv3 for IPv6 Commands** for OSPFv3 Authentication and Encryption command syntaxes, parameters, descriptions, and command examples.

OSPFv3 Authentication and Encryption Support

The IPv6 ESP header may be applied alone or in combination with the IPv6 AH header. When IPv6 ESP is used, both encryption and authentication are provided.

- Apply the IPsec AH header for OSPFv3 authentication using the **ipv6 ospf authentication spi** command.
- Apply the IPsec ESP header for OSPFv3 encryption using the **ipv6 ospf encryption spi esp** command.
- Confirm OSPFv3 authentication and encryption configuration using the **show ipv6 ospf** and **show debugging ipv6 ospf** commands.

When you only configure the IPv6 ESP header alone with the **ipv6 ospf encryption spi esp** command, note that both encryption and authentication are provided in IPsec ESP headers. You can apply IPv6 ESP headers alone or in combination with IPv6 AH headers.

If you configure IPv6 AH or IPv6 ESP headers for an OSPFv3 area, note that authentication or encryption is applied to all of the interfaces in the OSPFv3 area, except for any interfaces which had direct configuration of OSPFv3 authentication and/or encryption.

The below Authentication algorithms are supported in AlliedWare Plus OSPFv3 Authentication as parameter options in the **ipv6 ospf authentication spi** command:

- **MD5** (Message Digest Algorithm 5)
- **SHA-1** (Secure Hash Algorithm 1)

The below Encryption algorithms are supported in the AlliedWare Plus OSPFv3 Encryption as parameter options the **ipv6 ospf encryption spi esp** command:

- **AES** (Advanced Encryption Standard)
- **3DES** (Triple Data Encryption Standard)
- **NULL** (ESP without encryption)

See the sections listed below when configuring OSPFv3 Authentication and Encryption:

- **Configuring OSPFv3 Authentication on a VLAN**
- **Configuring OSPFv3 Encryption on a VLAN**
- **Configuring OSPFv3 Authentication in an OSPFv3 Area**
- **Configuring OSPFv3 Encryption in an OSPFv3 Area**

See sample configurations listed below with topologies that accompany these sections:

- **OSPFv3 Authentication in an OSPFv3 Area**
- **OSPFv3 Encryption in an OSPFv3 Area**
- **OSPFv3 Authentication on a VLAN**
- **OSPFv3 Encryption on a VLAN**
- **OSPFv3 Authentication with Two VLANs**
- **OSPFv3 Encryption with Two VLANs**

OSPFv3 Virtual Links

OSPFv3 Authentication and Encryption on virtual links requires some restriction on how the source IPv6 address that is used on packets sent over the virtual link is chosen.

For non-virtual links, IPv6 link-local addresses are used in packet headers. For virtual links, OSPFv3 uses non-link-local addresses, since packets are routed over intermediate routers.

A router does not necessarily have just one IPv6 address to choose from when deciding the source IPv6 address to use. A router may have more than one IPv6 address on the interfaces that connect it to the virtual link's transit area.

Without OSPFv3 Authentication and Encryption, multiple IPv6 addresses are not a problem. But multiple IPv6 addresses are a problem with OSPFv3 Authentication and Encryption, because the calculation of security parameters depends on knowing the addresses being used on both ends of the link.

Before IPv6 packets are exchanged over an OSPFv3 virtual link, the security parameters for the virtual link must be calculated. To calculate the security parameters, the router must know the source and destination addresses that will be used for OSPFv3 packets sent over the link.

A router knows what source IPv6 address it will use, and it knows the destination IPv6 address that it will send the packets to, since RFC 2740 (OSPF for IPv6) defines which of a virtual neighbor's addresses to use. However, a router cannot be sure what IPv6 address its virtual neighbor will use, since RFC 2740 does not specify the choice of source IPv6 address.

So to remove any uncertainty, routers that implement OSPFv3 Authentication and Encryption over virtual links ensure that the source IPv6 address of the packets they send over the virtual link is the same IPv6 address that their neighbor will use as a destination IPv6 address for its packets. Then IPv6 addresses that will be used over the virtual link are predictable. So, the security parameters can be calculated before the packet exchange begins.

If the IPv6 address at either end of the OSPFv3 virtual link changes, then the existing security parameters are discarded, before new security parameters are calculated.

- Apply the OSPFv3 authentication virtual link authentication using the **area virtual-link authentication ipsec spi** command.
- Apply the OSPFv3 virtual link encryption using the **area virtual-link encryption ipsec spi** command.
- Confirm OSPFv3 virtual link authentication and encryption configuration using the **show ipv6 ospf virtual-links** command.

See the section listed below when configuring OSPFv3 Virtual Links:

- **Configuring OSPFv3 Authentication and Encryption for a Virtual Link**

See sample configurations listed below with topologies that accompany this section:

- **OSPFv3 Authentication for a Virtual Link**
- **OSPFv3 Encryption for a Virtual Link**

Configuring OSPFv3

The following examples show firstly the configuration for a basic three-switch single-area OSPFv3 network. This network is then modified to become a two-area network.

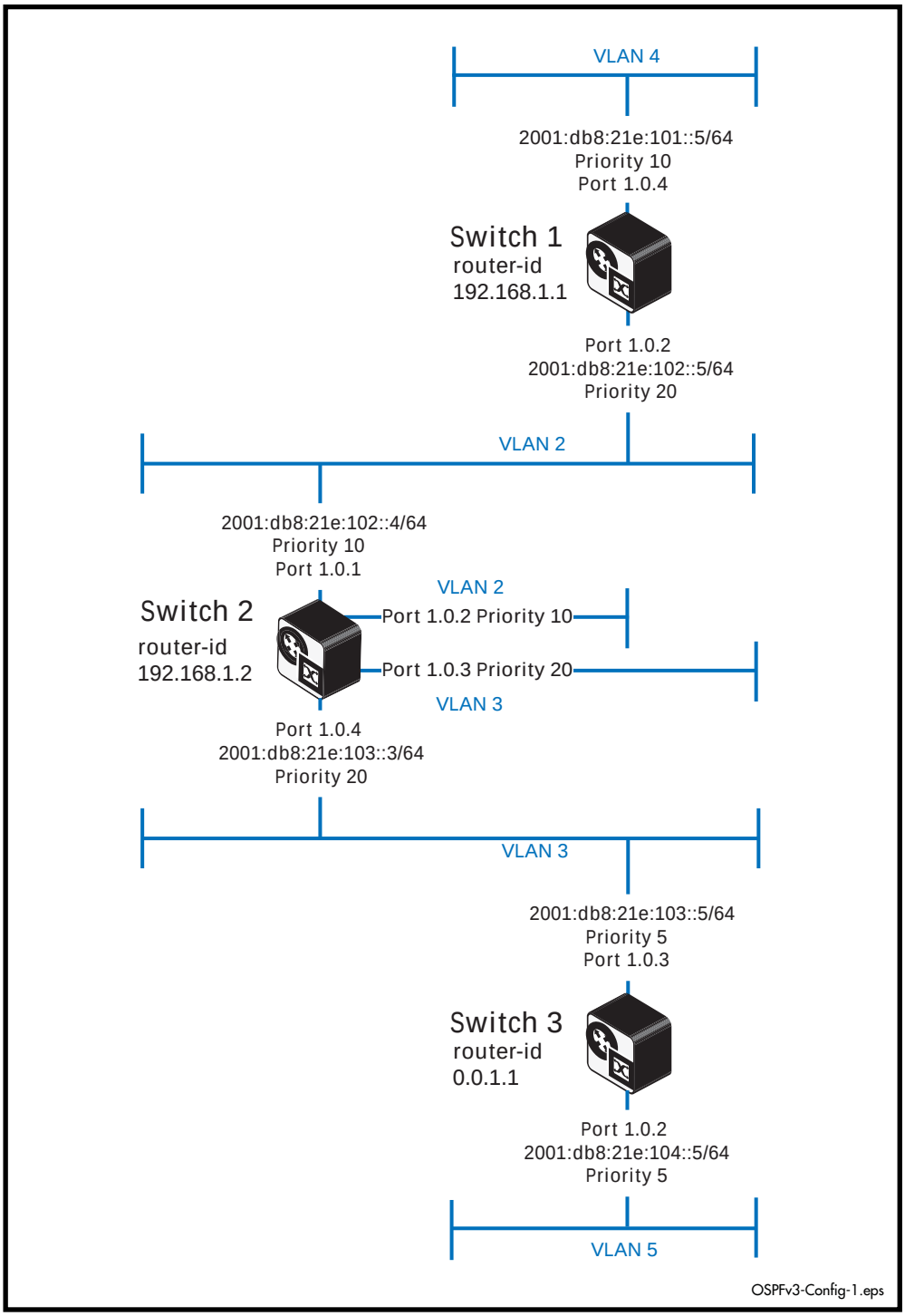
The configuration command sequences shown are presented for understanding, and do not necessarily represent the most efficient configuration method.

The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

Example 1: Single-Area Network Configuration

Figure 43-8 shows a basic three-switch, single-area configuration. Setting the priority value of 20 to VLAN 2 on switch 1 and to the value 10 on switch 2 will ensure that switch 1 will become the designated router for this VLAN. Similarly, the priority settings on VLAN 3, of 20 on switch 2, and 5 on switch 3, will ensure that switch 2 becomes the designated router for VLAN 3.

Figure 43-8: Basic single-area OSPFv3 configuration



Switch 1 configuration (single-area network)

Table 43-4: Example 1—single-area network configuration—Switch 1

Step 1. Create and enable VLANs	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter VLAN database mode.
`awplus(config-vlan)#`	
`vlan 2,4 state enable`	Create and enable VLANs 2 and 4.
`awplus(config-vlan)#`	
`exit`	Return to Global Configuration mode.
Step 2. For VLAN 4, assign IPv6 addresses and configure OSPF	
`awplus(config)#`	
`interface vlan4`	Select VLAN 4 for configuring.
`awplus(config-if)#`	
`ipv6 address 2001:db8:21e:101::5/64`	Assign an IPv6 address to VLAN 4.
`awplus(config-if)#`	
`ipv6 router ospf area 0 tag process1`	Assign VLAN 4 to OSPF area 0. In this configuration, any address within the subnet 2001:db8:21e:101::/64 will be part of OSPF area 0.
Note that the tag named "process1" denotes a separate router process. Its name can comprise any string of alphanumeric characters. Also note that this tag is local to the router on which it is set and does not appear in any OSPF packets or LSAs. For more information on processes and instances, see **"Configuring instances and processes" on page 43.31**.	
`awplus(config-if)#`	
`ipv6 ospf priority 10`	Set the priority for the router to become the designated router for a particular VLAN. The router with the highest priority becomes the designated router for that VLAN. Defaults to 1.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
Step 3. For VLAN 2, assign IPv6 addresses and configure OSPF	
`awplus(config)#`	
`interface vlan2`	Select VLAN 2 for configuring.
`awplus(config-if)#`	
`ipv6 address 2001:db8:21e:102::5/64`	Assign an IPv6 address to VLAN 2.
`awplus(config-if)#`	
`ipv6 router ospf area 0 tag process1`	Assign VLAN 2 to OSPF area 0. In this configuration, any address within the subnet 2001:db8:21e:102::/64 will be part of OSPF area 0.

Table 43-4: Example 1—single-area network configuration—Switch 1(cont.)

`awplus(config-if)#`	
`ipv6 ospf priority 20`	Set the priority for the router to become the designated router for a particular VLAN. The router with the highest priority becomes the designated router for that VLAN. Defaults to 1.
Step 4. Configure switch ports and assign them to their VLANs	
`awplus(config)#`	
`interface port1.0.4`	Select port 1.0.4 for configuring.
`awplus(config-if)#`	
`switchport access vlan 4`	Assign port 1.0.4 to VLAN 4.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
`awplus(config)#`	
`interface port1.0.2`	Select port1.0.2 for configuring.
`awplus(config-if)#`	
`switchport access vlan 2`	Assign port 1.0.2 to VLAN 2.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
Step 5. Set router ID and allocate memory to IPv6	
`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router ipv6 ospf process1`	Configure an IPv6 OSPF routing process.
`awplus(config-router)#`	
`router-id 192.168.1.1`	Set the router ID.
`awplus(config-router)#`	
`passive-interface vlan4`	Configure VLAN 4 as a passive interface to suppress IPv6 OSPF routing updates.
`awplus(config-router)#`	
`exit`	Return to Global Configuration mode.
`awplus(config)#`	
`ipv6 forwarding`	Turn on IPv6 forwarding.
`awplus(config)#`	
`exit`	Return to Privileged Exec mode.

Switch 2 configuration (single-area network)

Table 43-5: Example 1—single-area network configuration—Switch 2

Step 1. Create and enable VLANs 2 and 3	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter VLAN database mode.
`awplus(config-vlan)#`	
`vlan 2,3 state enable`	Create and enable VLANs 2 and 3.
`awplus(config-vlan)#`	
`exit`	Return to Global Configuration mode.
Step 2. For VLAN 2, assign IPv6 addresses and configure OSPF	
`awplus(config)#`	
`interface vlan2`	Select VLAN 2 for configuring.
`awplus(config-if)#`	
`ipv6 address 2001:db8:21e:102::4/64`	Assign an IPv6 address to VLAN 2.
`awplus(config-if)#`	
`ipv6 router ospf area 0 tag process1`	Assign VLAN 2 to OSPF area 0. In this configuration, any address within the subnet 2001:db8:21e:102 will be part of OSPF area 0.
`awplus(config-if)#`	
`ipv6 ospf priority 10`	Set the priority for the router to become the designated router for a particular VLAN. The router with the highest priority becomes the designated router for that VLAN. Defaults to 1.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
Step 3. For VLAN 3, assign IPv6 addresses and configure OSPF	
`awplus(config)#`	
`interface vlan3`	Select VLAN 3 for configuring.
`awplus(config-if)#`	
`ipv6 address 2001:db8:21e:103::3/64`	Assign an IPv6 address to VLAN 3.
`awplus(config-if)#`	
`ipv6 router ospf area 0 tag process1`	Assign VLAN 3 to OSPF area 0. In this configuration, any address within the subnet 2001:db8:21e:103 will be part of OSPF area 0.
`awplus(config-if)#`	
`ipv6 ospf priority 20`	Set the priority for the router to become the designated router for a particular VLAN. The router with the highest priority becomes the designated router for that VLAN. Defaults to 1.

Table 43-5: Example 1—single-area network configuration—Switch 2(cont.)

Step 4. Configure switch ports and assign them to their VLANs	
`awplus(config)#`	
`interface port1.0.1`	Select port 1.0.1 for configuring.
`awplus(config-if)#`	
`switchport access vlan 2`	Assign port 1.0.1 to VLAN 2.
`awplus(config-if)#`	
`exit`	Return to Global Config mode.
`awplus(config)#`	
`interface port1.0.2`	Select port1.0.2 for configuring.
`awplus(config-if)#`	
`switchport access vlan 2`	Assign port 1.0.2 to VLAN 2.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
`awplus(config)#`	
`interface port1.0.3`	Select port1.0.3 for configuring.
`awplus(config-if)#`	
`switchport access vlan 3`	Assign port 1.0.3 to VLAN 3.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
`awplus(config)#`	
`interface port1.0.4`	Select port1.0.4 for configuring.
`awplus(config-if)#`	
`switchport access vlan 3`	Assign port 1.0.4 to VLAN 3.
Step 5. Set router ID and allocate memory to IPv6	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`router ipv6 ospf process1`	Configure an IPv6 routing process.
`awplus(config-router)#`	Set the router ID. Note that:
`router-id 192.168.1.2`	■ In this configuration, the router-id value 192.168.1.2 makes this switch most likely to be selected as the designated router.
`awplus(config)#`	
`ipv6 forwarding`	Turn on IPv6 forwarding.
`awplus(config)#`	
`exit`	Return to Privileged Exec mode.

Switch 3 configuration (single-area network)

Table 43-6: Example 1—single-area network configuration—Switch 3

Step 1. Create and enable VLANs 3 and 5	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter VLAN database mode.
`awplus(config-vlan)#`	
`vlan 3,5 state enable`	Create and enable VLANs 3 and 5.
`awplus(config-vlan)#`	
`exit`	Return to Global Configuration mode.
Step 2. For VLAN 3, assign IPv6 addresses and configure OSPF	
`awplus(config)#`	
`interface vlan 3`	Select VLAN 3 for configuring.
`awplus(config-if)#`	
`ipv6 address 2001:db8:21e:103::5/64`	Assign an IPv6 address to VLAN 3.
`awplus(config-if)#`	
`ipv6 router ospf area 0 tag process1`	Assign VLAN 3 to OSPF area 0. In this configuration, any address within the subnet 2001:db8:21e:103::/64 will be part of OSPF area 0.
`awplus(config-if)#`	
`ipv6 ospf priority 5`	Set the priority for the router to become the designated router for a particular VLAN. The router with the highest priority becomes the designated router for that VLAN. Defaults to 1.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
Step 3. For VLAN 5, assign IPv6 addresses and configure OSPF	
`awplus(config)#`	
`interface vlan5`	Select VLAN 5 for configuring.
`awplus(config-if)#`	
`ipv6 address 2001:db8:21e:104::5/64`	Assign an IPv6 address to VLAN 5.
`awplus(config-if)#`	
`ipv6 router ospf area 0 tag process1`	Assign VLAN 5 to OSPF area 0. In this configuration, any address within the subnet 2001:db8:21e:104::/64 will be part of OSPF area 0.
`awplus(config-if)#`	
`ipv6 ospf priority 5`	Set the priority for the router to become the designated router for a particular VLAN. The router with the highest priority becomes the designated router for that VLAN. Defaults to 1.

Table 43-6: Example 1—single-area network configuration—Switch 3(cont.)

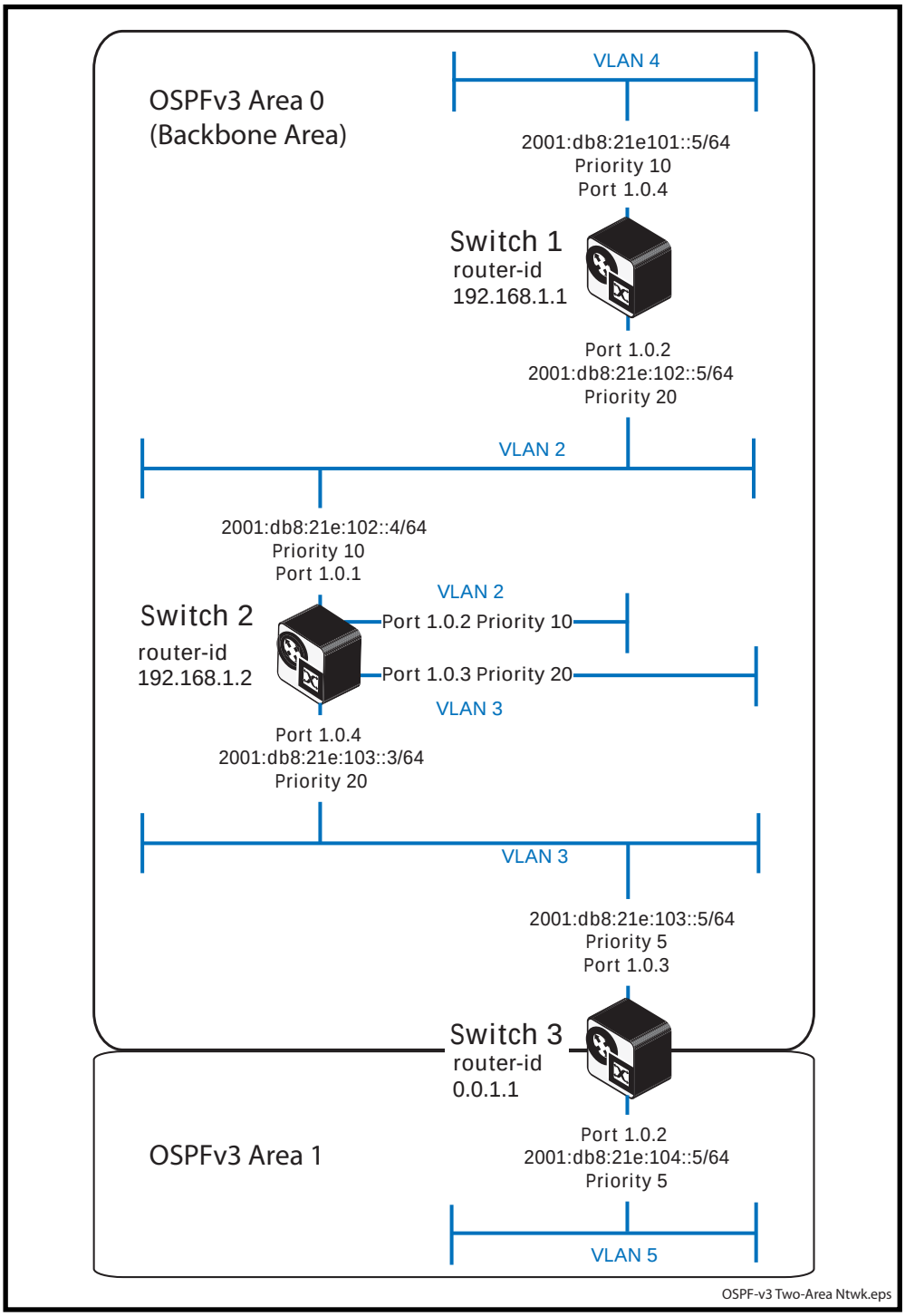
Step 4. Configure switch ports and assign them to their VLANs	
`awplus(config)#`	
`interface port1.0.3`	Select port 1.0.3 for configuring.
`awplus(config-if)#`	
`switchport access vlan 3`	Assign port 1.0.3 to VLAN 3
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
`awplus(config)#`	
`interface port1.0.2`	Select port1.0.2 for configuring.
`awplus(config-if)#`	
`switchport access vlan 5`	Assign port 1.0.2 to VLAN 5.
`awplus(config-if)#`	
`exit`	Return to Global Configuration mode.
Step 5. Set router ID and allocate memory to IPv6	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`router ipv6 ospf process1`	Configure an IPV6 routing process.
`awplus(config-router)#`	
`router-id 0.0.1.1`	Set the router ID. Note that the router-id value 0.0.1.1 has been selected for the following reasons: ■ so that it is unlikely to be selected as a designated router. ■ to illustrate that it, although it has an IPv4 address format, it is not an IPv4 address.
`awplus(config-router)#`	
`passive-interface vlan5`	Configure VLAN 5 as a passive interface to suppress IPv6 OSPF routing updates.
`awplus(config-router)#`	
`exit`	Return to Global Configuration mode.
`awplus(config)#`	
`ipv6 forwarding`	Turn on IPv6 forwarding.
`awplus(config)#`	
`exit`	Return to Privileged Exec mode.

Example 2: Two-Area Network Configuration

Figure 43-9 shows a basic three-switch, two-area configuration. Setting the priority value of 20 to VLAN 2 on Switch 1 and to the value 10 on Switch 2 will ensure that Switch 1 will become the designated router for this VLAN. Similarly the priority settings on VLAN 3 of 20 on Switch 2, and 5 on Switch 3 will ensure that Switch 2 will become the designated router for VLAN 3.

Note that Switch 3 now becomes the area border router for areas 0 and 1.

Figure 43-9: Basic two-area OSPFv3 configuration



The commands used to configure this two-area network are the same as those used in the previous example, except for the configuration of VLAN 5 on switch 3, which now becomes part of Area 1. The revised configuration for switch 3 and VLAN 5 is shown in below.

Table 43-7: Example 2—two-area network—revised configuration for Switch 3

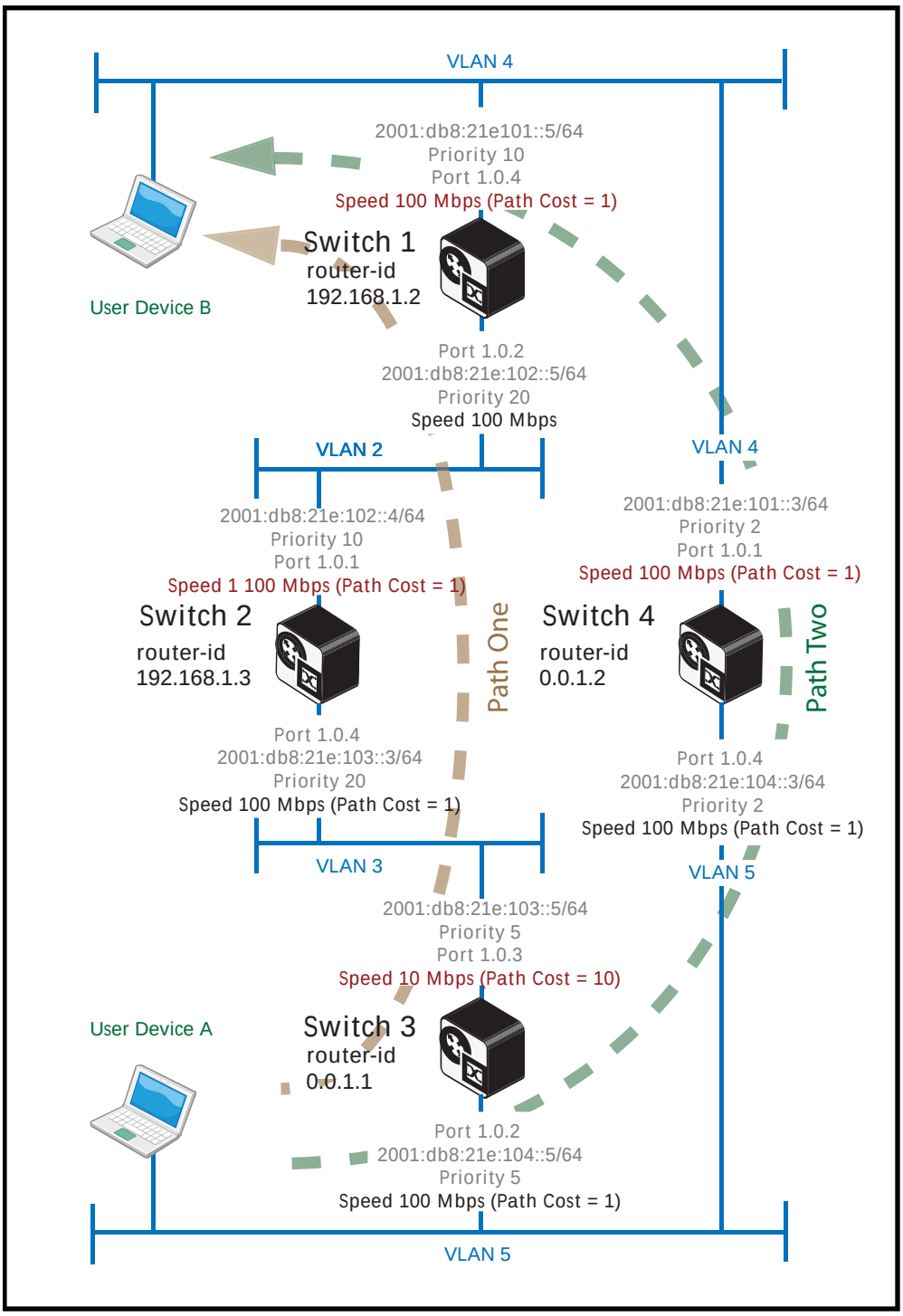
Set VLANs 3 and 5 to be part of area 1	
`awplus(config)#`	
`interface vlan5`	Select VLAN 5 for configuring.
`awplus(config-if#`	
`ipv6 address 2001:db8:21e:104::5/64`	Assign IPv6 address to VLAN 5.
`awplus(config-if)#`	
`ipv6 router ospf area 1`	Assign VLAN 5 to OSPF area 1. In this configuration, any address within the subnet 2001:db8:21e:104 will be part of OSPF area 1.

Setting Up the Metrics

OSPF selects its preferred routes by measuring the cost of each path to a given destination. Each path (or link) has a metric value applied, which by default is 100 000 000 divided by the link bandwidth. The preferred route will be the one that presents the lowest total path cost.

Figure 43-10 shows the previously configured single-area network of **Figure 43-8**, but with an additional path provided by Switch 4. The diagram also shows port speeds and their resultant path costs.

Figure 43-10: OSPFv3 multipath metrics example



Using the procedure shown in **"Switch 1 configuration (single-area network)" on page 43.20**, add Switch 4 to the configuration, applying the port numbers, VLANs, and settings shown in **Figure 43-10 on page 43.28**.

Setting the metric values

The metric values shown are derived from the data rate configured for each port and (unless manually changed) use the formula 10^8 / port bandwidth in bps.

Example To set the speed of a tri-speed port to 100 Mbps, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# speed 100
```

The above example will produce a path cost of 100 000 000 / port speed (in bps), or 100 000 000 /100 000 000 = 1

If the port speed and its resultant metric value does not produce the required best paths and you do not want to change the port speed, you can change the metric by using the **ipv6 ospf cost** command on page 44.35. The following section explains how to do this.

Best path selection

Figure 43-10 on page 43.28 shows two paths from User Device A to User Device B, labeled Path One and Path Two. The metric cost for each of these paths is found by adding the individual output path costs: For traffic flowing from User Device A, to User Device B, these are:

Path Cost for Path One = 10 + 1 + 1 = 12

Path Cost for Path Two = 1

The preferred route will be via the paths that offer the lowest total path cost. Traffic will therefore travel from User Device A, to User Device B via Path Two.

However, you can manually adjust the cost metrics to favor one path over another. For example, you can apply the **ipv6 ospf cost** command on page 44.35 to manually force the preference for a particular path.

Example Using the configuration shown in **Figure 43-10 on page 43.28** use the **ipv6 ospf cost** command on page 44.35 set Path One to be the preferred path.

Existing path cost for Path One is, 10 + 1 + 1 = 12

Existing path cost for Path Two = 1

Setting the cost of Path Two to be greater than 12, will force OSPF's path selection algorithm to select Path One as the preferred path between User Device A and User Device B.

Use the following commands on switch 4 to force the path cost on switchport 1.0.1 to have the metric value of 20:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# ipv6 ospf 2001:0db8:21e:101::5/64 cost 20
```

This has set a higher metric cost for data traveling from Device A to Device B, via Path Two. You now need to set the same metric cost for data traveling in the direction from Device B, to Device A.

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ipv6 ospf 2001:0db8:21e:104::5/64 cost 20
```

Traffic traveling from devices A to B, and B to A, will now use Path One in preference to Path Two.

Configuring instances and processes

OSPFv3 introduces the capability of sharing physical links (or VLANs) with multiple OSPF routing instances. This is achieved by inserting an Instance ID field in the OSPFv3 **Hello**, and **Link State** packets, see **"OSPFv3 packet header" on page 43.5**. In a multi-instance network, the OSPF hello and LSA frames belonging to different instances can coexist alongside each other, each sharing a physical link and each establishing its own set of metric values and path preferences. New instances are created using the **ipv6 router ospf area** command on page 44.46.

There are solutions where multiple OSPF routers are attached to a single link, but should not form an adjacency between OSPF routers. For example, four OSPF routers are attached to an Ethernet link. OSPF routers 1 and 2 belong to one OSPF domain, and OSPF routers 3 and 4 belong to a different OSPF domain. There should be adjacencies between OSPF routers 1 and 2, and between OSPF routers 3 and 4. But there should not be adjacencies between OSPF routers 1 and 3 or between OSPF routers 2 and 4 in this example.

This type of separation of adjacencies is often accomplished in the case of OSPFv2 by the use of manipulated OSPF authentication. However, this is not ideal since OSPF routers may log authentication failures of any rejected hello packets associated with the OSPF domain.

OSPFv3 allows for multiple OSPF routing instances per link by adding the new Instance ID field to the OSPF packet header to distinguish OSPF routing instances. An interface assigned to a given interface ID will drop OSPF packets whose Instance IDs do not match.

Some OSPFv2 (and OSPFv3) switches and routers also have a similar (and older) function called processes. Processes enable more than one OSPF environment to be configured on the same switch (or router), but not the same VLAN. Processes exist only within the switch or router on which they are created. Although their existence is not explicitly conveyed between OSPF routers, the packets from each process contains its own router ID and will thus appear as if generated by a different OSPF router. New processes are created also using the **ipv6 router ospf area** command on page 44.46.

Although instances and processes apply a similar function and each can be individually created, there are interactions between them that require you to apply some specific rules when creating multi-instance networks. The following table shows those combinations of instances and processes that are workable, and those that are not.

Table 43-8: Functional combinations of OSPFv3 instances and processes

VLAN	Instances	Processes	Workable Combination
VLAN 5	Instance 1	Process A	Yes
VLAN 5 & VLAN 7	Instance 1	Process D	Yes
VLAN 5	Instance 1	Process D	Yes
If you now add a second instance so that VLAN 5 shares both instances using Process D			
VLAN 5	Instance 2	Process D	No
To make the above configuration work requires adding a new process.			
VLAN 5	Instance 1	Process D	Yes
VLAN 5	Instance 2	Process E	Yes

Example 1 To apply the settings shown in the last two rows of **Table 43-8**, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ipv6 router ospf area 1 tag D instance-id 1
awplus(config-if)# ipv6 router ospf area 2 tag E instance-id 2
```

Example 2 See the below entries for another supported combination of OSPFv3 instances and processes:

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ipv6 router ospf area 1 tag D instance-id 1
awplus(config-if)# ipv6 router ospf area 1 tag E instance-id 2
awplus(config-if)# ipv6 router ospf area 2 tag F instance-id 3
awplus(config-if)# exit
awplus(config)# interface vlan7
awplus(config-if)# ipv6 router ospf area 1 tag D instance-id 1
```

Example 3 See the below entries for an unsupported combination of OSPFv3 instances and processes. Note the console error message shown after completing the entries with the same instance ID (**1**) on different processes (**D** and **E**) on the same interface (**vlan5**):

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ipv6 router ospf area 1 tag D instance-id 1
awplus(config-if)# ipv6 router ospf area 1 tag E instance-id 1
% Interface enabled in another process with
the same instance ID
```

Example 4 See the below entries for an unsupported combination of OSPFv3 instances and processes. Note the console error message shown after completing the entries with different instances IDs (**1** and **2**) on the same process (**A**) on the same interface (**vlan5**):

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ipv6 router ospf area 1 tag A instance-id 1
awplus(config-if)# ipv6 router ospf area 2 tag A instance-id 2
% Interface enabled in the same process with
a different instance ID
```

Configuring OSPFv3 Authentication and Encryption

See the section **OSPFv3 Authentication and Encryption Overview** earlier in this chapter for introductory OSPFv3 Authentication and OSPFv3 Encryption information.

Also refer to **Chapter 44, OSPFv3 for IPv6 Commands** for detailed OSPFv3 Authentication and OSPFv3 Encryption command syntaxes, parameters, descriptions, and command examples.

Once you configure OSPFv3 and decide on using authentication or encryption, you need to define a security policy on each OSPFv3 interface of each device in the network.

A security policy consists of:

- An SPI (Security Parameter Index), the identification number for the security policy
- An authentication algorithm, either MD5 (Message Digest 5) or SHA-1 (Secure Hash Algorithm 1) authentication, if the security policy is to perform authentication
- An authentication key, which is a 32 or 40 bit hexadecimal value used with the authentication algorithm, if the security policy is to perform authentication
- An encryption algorithm, either AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) or 3DES (Triple Data Encryption Standard) encryption, if the security policy is to perform encryption.
- An encryption key, which is a 32 to 64 bit hexadecimal value used with the encryption algorithm, if the security policy is to perform encryption

You can configure authentication and encryption on virtual links, on VLAN interfaces, and on OSPFv3 areas. When you configure authentication or encryption for an area, the security policy is applied to all of the VLAN interfaces in the OSPFv3 area. Note that you should use a different security policy on each VLAN interface to improve the security.

See the sections listed below when configuring OSPFv3 Authentication:

- **Configuring OSPFv3 Authentication on a VLAN**
- **Configuring OSPFv3 Encryption on a VLAN**
- **Configuring OSPFv3 Authentication in an OSPFv3 Area**
- **Configuring OSPFv3 Encryption in an OSPFv3 Area**
- **Configuring OSPFv3 Authentication and Encryption for a Virtual Link**

See sample configurations listed below with topologies that accompany these sections:

- **OSPFv3 Authentication in an OSPFv3 Area**
- **OSPFv3 Encryption in an OSPFv3 Area**
- **OSPFv3 Authentication on a VLAN**
- **OSPFv3 Encryption on a VLAN**
- **OSPFv3 Authentication with Two VLANs**
- **OSPFv3 Encryption with Two VLANs**
- **OSPFv3 Authentication for a Virtual Link**
- **OSPFv3 Encryption for a Virtual Link**

Configuring OSPFv3 Authentication on a VLAN

Enter the below commands to configure OSPFv3 Authentication on a VLAN interface.

Note you need to configure OSPFv3 on a VLAN before configuring OSPFv3 Authentication.

Note the links in the list are for the command name in the relevant command chapter. The links in the table below the list take you to the command syntax of the relevant command.

Commands Applied

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **interface (to configure)**
4. **ipv6 ospf authentication spi**
5. **exit**
6. **exit**

Table 43-9: Configuring the DHCPv6 Configuration Pool

1.	`awplus>enable [<privilege-level>]`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enter Interface Configuration mode.
4.	`awplus(config-if)#ipv6 ospf authentication ipsec spi <256-4294967295> {md5 <MD5-key>|sha1 <SHA1-key>}`	Specifies an OSPFv3 Authentication type (MD5 or SHA-1) for a specified VLAN interface.
5.	`awplus(config-if)# exit`	Return to Global Configuration mode.
6.	`awplus(config)# exit`	Return to Privileged Exec mode.

Configuring OSPFv3 Encryption on a VLAN

Enter the below commands to configure OSPFv3 Encryption on a VLAN interface.

Note you need to configure OSPFv3 on a VLAN before configuring OSPFv3 Encryption.

Note the links in the list are for the command name in the relevant command chapter. The links in the table below the list take you to the command syntax of the relevant command.

Commands Applied

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **interface (to configure)**
4. **ipv6 ospf encryption spi esp**
5. **exit**
6. **exit**

Table 43-10: Configuring the DHCPv6 Configuration Pool

1.	`awplus>enable (Privileged Exec mode)`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enter Interface Configuration mode.
4.	`awplus(config-if)#ipv6 ospf encryption ipsec spi <256-4294967295> esp {aes-cbc <AES-CBC-key>|3des <3DES-key>|null}{md5 <MD5-key>|sha1 <SHA1-key>}`	Specifies an OSPFv3 Encryption type (AES-CBC or 3DES) and an OSPFv3 Authentication type (MD5 or SHA-1) for a specified VLAN interface.
5.	`awplus(config-if)# exit`	Return to Global Configuration mode.
6.	`awplus(config)# exit`	Return to Privileged Exec mode.

Configuring OSPFv3 Authentication in an OSPFv3 Area

Enter the below commands to configure OSPFv3 Authentication in an OSPFv3 Area.

Note you need to configure an OSPFv3 Area before configuring OSPFv3 Authentication.

Note the links in the list are for the command name in the relevant command chapter. The links in the table below the list take you to the command syntax of the relevant command.

Commands Applied

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **interface (to configure)**
4. **ipv6 router ospf area**
5. **area authentication ipsec spi**
6. **exit**
7. **exit**
8. **exit**

Table 43-11: Configuring the DHCPv6 Configuration Pool

1.	`awplus>enable (Privileged Exec mode)`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enter Interface Configuration mode.
4.	`awplus(config-if)# ipv6 router ospf area <area-id> [tag <process-id>] [instance <inst-id>]`	Specify an OSPFv3 Area and, and enter Router Configuration mode.
5.	`awplus(config-router)# area <area-id> authentication ipsec spi <256-4294967295> {md5 <MD5-key>|sha1 <SHA1-key>}`	Specifies an OSPFv3 Authentication type (MD5 or SHA-1) for a specified OSPFv3 Area.
6.	`awplus(config-router)# exit`	Return to Interface Configuration mode.
7.	`awplus(config-if)# exit`	Return to Global Configuration mode.
8.	`awplus(config)# exit`	Return to Privileged Exec mode.

Configuring OSPFv3 Encryption in an OSPFv3 Area

Enter the below commands to configure OSPFv3 Encryption in an OSPFv3 Area.

Note you need to configure an OSPFv3 Area before configuring OSPFv3 Encryption.

Note the links in the list are for the command name in the relevant command chapter. The links in the table below the list take you to the command syntax of the relevant command.

Commands Applied

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **interface (to configure)**
4. **ipv6 router ospf area**
5. **area encryption ipsec spi esp**
6. **exit**
7. **exit**
8. **exit**

Table 43-12: Configuring the DHCPv6 Configuration Pool

1.	`awplus>enable (Privileged Exec mode)`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enter Interface Configuration mode.
4.	`awplus(config-if)# ipv6 router ospf area <area-id> [tag <process-id>] [instance <inst-id>]`	Specify an OSPFv3 Area and, and enter Router Configuration mode.
5.	`awplus(config-router)# area <area-id> encryption ipsec spi <256-4294967295> esp {aes-cbc <AES-CBC-key>| 3des <3DES-key>|null}{md5 <MD5-key>| sha1 <SHA1-key>}`	Specifies an OSPFv3 Encryption type (AES-CBC or 3DES) and an OSPFv3 Authentication type (MD5 or SHA-1) for a specified OSPFv3 Area.
6.	`awplus(config-router)# exit`	Return to Interface Configuration mode.
7.	`awplus(config-if)# exit`	Return to Global Configuration mode.
8.	`awplus(config)# exit`	Return to Privileged Exec mode.

Configuring OSPFv3 Authentication and Encryption for a Virtual Link

Enter the below commands to configure OSPFv3 Authentication and OSPFv3 Encryption for a OSPFv3 Area Virtual Link. Note you need to configure an OSPFv3 Area and an OSPFv3 Area Virtual Link before configuring OSPFv3 Authentication and OSPFv3 Encryption for an OSPFv3 Area Virtual Link.

Note the links in the list are for the command name in the relevant command chapter. The links in the table below the list take you to the command syntax of the relevant command.

Commands Applied

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **interface (to configure)**
4. **ipv6 router ospf area**
5. **area virtual-link (IPv6 OSPF)**
6. **area virtual-link authentication ipsec spi**
7. **area virtual-link encryption ipsec spi**
8. **exit**
9. **exit**
10. **exit**

Table 43-13: Configuring the DHCPv6 Configuration Pool

1.	`awplus>enable (Privileged Exec mode)`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enter Interface Configuration mode.
4.	`awplus(config-if)# ipv6 router ospf area <area-id> [tag <process-id>] [instance <inst-id>]`	Specify an OSPFv3 Area and, and enter Router Configuration mode.
5.	`awplus(config-router)# area <area-id> virtual-link <router-id>`	Specifies an OSPFv3 Area Virtual Link.
6.	`awplus(config-router)# area <area-id> virtual-link <router-ID> authentication ipsec spi <256-4294967295> {md5 <MD5-key>|sha1 <SHA1-key>}`	Specifies an OSPFv3 Encryption type (AES-CBC or 3DES) and an OSPFv3 Authentication type (MD5 or SHA-1) for a specified OSPFv3 Area.

Table 43-13: Configuring the DHCPv6 Configuration Pool

7.	`awplus(config-router)# area <area-id> virtual-link <router-ID> encryption ipsec spi <256-4294967295> esp {aes-cbc <AES-CBC-key>|3des <3DES-key>| null}{md5 <MD5-key>|sha1 <SHA1-key>}`	Specifies an OSPFv3 Encryption type (AES-CBC or 3DES) and an OSPFv3 Authentication type (MD5 or SHA-1) for a specified OSPFv3 Area.
8.	`awplus(config-router)# exit`	Return to Interface Configuration mode.
9.	`awplus(config-if)# exit`	Return to Global Configuration mode.
10.	`awplus(config)# exit`	Return to Privileged Exec mode.

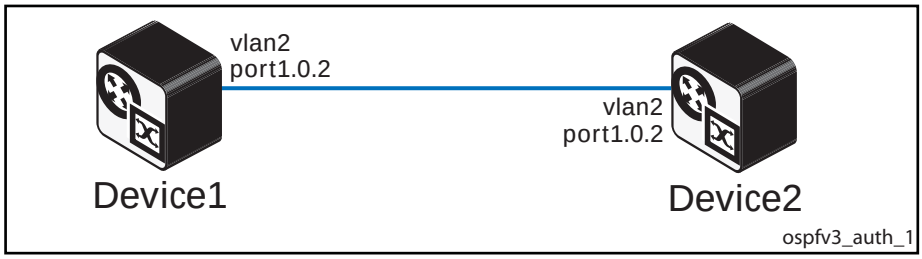
OSPFv3 Authentication in an OSPFv3 Area

This example shows how to configure two devices to form an adjacency to each other with OSPFv3 authentication in an OSPFv3 backbone area 0. Both devices are configured to use the same process name test1, which is not advertised between peers.

See section **Configuring OSPFv3 Authentication in an OSPFv3 Area** for the sequence of console command entries. For command information, see the **area authentication ipsec spi** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-11: OSPFv3 Authentication in an OSPFv3 Area topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device1_to_Device2
 ipv6 address 2001:db8:2::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.1
 area 0 authentication ipsec spi 256 sha1
12345678901234567890123456789012345678901
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device2_to_Device1
 ipv6 address 2001:db8:2::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
 area 0 authentication ipsec spi 256 sha1
12345678901234567890123456789012345678901
```

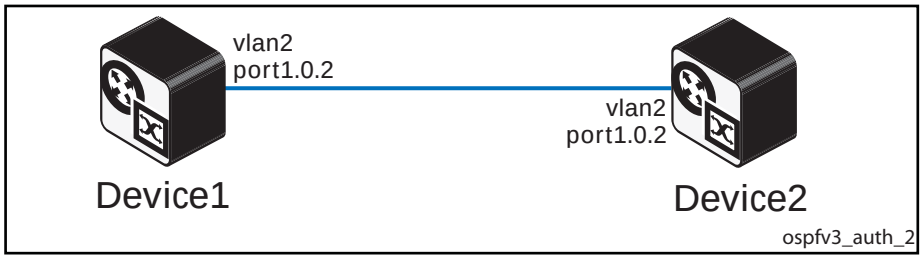
OSPFv3 Encryption in an OSPFv3 Area

This example shows how to configure two devices to form an adjacency to each other with OSPFv3 encryption in an OSPFv3 backbone area 0. Both devices are also configured to use the same internal process name test1. Additionally, both devices insert instance-id 1 into their advertised hello messages to control adjacency between the two devices.

See section **Configuring OSPFv3 Encryption in an OSPFv3 Area** for the sequence of console command entries. For command information, see the **area encryption ipsec spi esp** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-12: OSPFv3 Encryption in an OSPFv3 Area topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device1_to_Device2
 ipv6 address 2001:db8:2::1/64
 ipv6 router ospf area 0 tag test1 instance-id 1
!
ipv6 forwarding
!
router ipv6 ospf test1
 router-id 192.168.1.1
 area 0 encryption ipsec spi 555 esp aes-cbc
12345678901234567890123456789012 sha1
12345678900987654321123546789009876543 21
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device2_to_Device1
 ipv6 address 2001:db8:2::2/64
 ipv6 router ospf area 0 tag test1 instance-id 1
!
ipv6 forwarding
!
router ipv6 ospf test1
 router-id 192.168.1.2
 area 0 encryption ipsec spi 555 esp aes-cbc
12345678901234567890123456789012 sha1
12345678900987654321123546789009876543 21
```

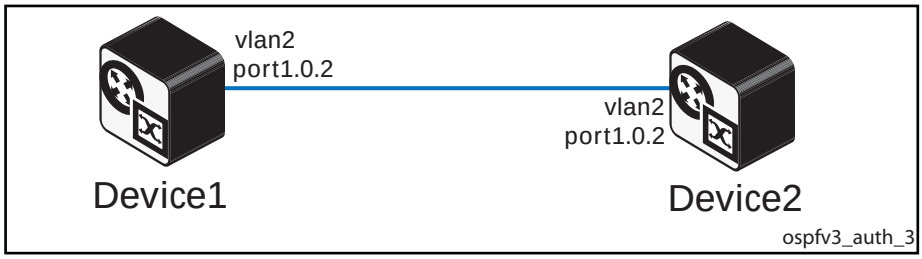
OSPFv3 Authentication on a VLAN

This example shows how to configure two devices with OSPFv3 authentication on a VLAN.

See section **Configuring OSPFv3 Authentication on a VLAN** for the sequence of console command entries. For detailed command information, see the **ipv6 ospf authentication spi** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-13: OSPFv3 Authentication on a VLAN topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device1_to_Device2
 ipv6 address 2001:db8:2::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf authentication ipsec spi 256 md5
1234567890abcdef1234567809abcdef
!
ipv6 forwarding
ipv6 ospf display route single-line
!
router ipv6 ospf process1
 router-id 192.168.1.1
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device2_to_Device1
 ipv6 address 2001:db8:2::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf authentication ipsec spi 256 md5
1234567890abcdef1234567809abcdef
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
```

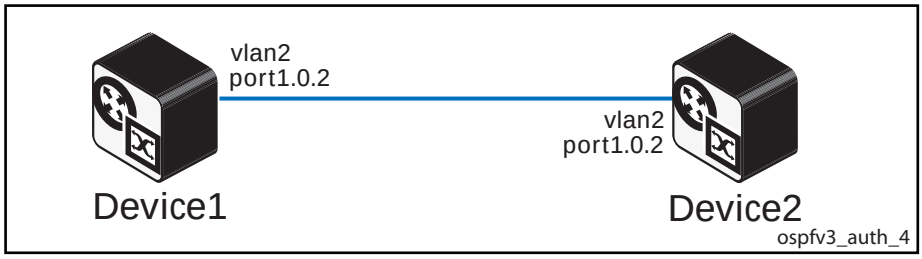
OSPFv3 Encryption on a VLAN

This example shows how to configure two devices with OSPFv3 encryption on a VLAN.

See section **Configuring OSPFv3 Encryption on a VLAN** for the sequence of console command entries. For detailed command information, see the **ipv6 ospf encryption spi esp** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-14: OSPFv3 Encryption on a VLAN topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device1_to_Device2
 ipv6 address 2001:db8:2::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf encryption ipsec spi 1000 esp 3des
1234567890abcdef1234567890abcdef12
!
ipv6 forwarding
ipv6 ospf display route single-line
!
router ipv6 ospf process1
 router-id 192.168.1.1
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device2_to_Device1
 ipv6 address 2001:db8:2::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf encryption ipsec spi 1000 esp 3des
1234567890abcdef1234567890abcdef12
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
```

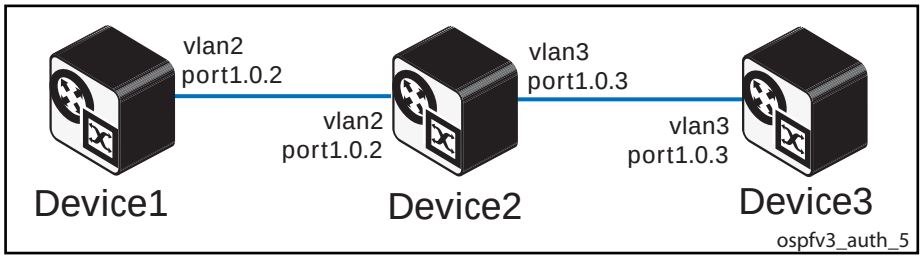
OSPFv3 Authentication with Two VLANs

This example shows how to configure three devices with OSPFv3 authentication on a per VLAN basis. Note that to increase security, rather than using a single SPI for the entire area, this example uses a different security profile for each VLAN in the topology shown.

See section **Configuring OSPFv3 Authentication on a VLAN** for the sequence of console command entries. For detailed command information, see the **ipv6 ospf authentication spi** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-15: OSPFv3 Authentication on a VLAN topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device1_to_Device2
 ipv6 address 2001:db8:2::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf authentication ipsec spi 256 md5
1234567890abcdef1234567809abcdef
!
ipv6 forwarding
ipv6 ospf display route single-line
!
router ipv6 ospf process1
 router-id 192.168.1.1
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2

vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!!
interface vlan2
 description Device2_to_Device1
 ipv6 address 2001:db8:2::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf authentication ipsec spi 256 md5
1234567890abcdef1234567809abcdef
!
interface vlan3
 description Device2_to_Device3
 ipv6 address 2001:db8:3::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf authentication ipsec spi 512 sha1
1234567890abcdef1234567890abcdef12345678
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
```

Device3 configuration

See the below configuration for a device with the hostname **Device3**:

```
hostname Device3
!
vlan database
 vlan 3 state enable
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!!
interface vlan3
 description Device3_to_Device2
 ipv6 address 2001:db8:3::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf authentication ipsec spi 512 sha1
1234567890abcdef1234567890abcdef12345678
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 3.3.3.3
```

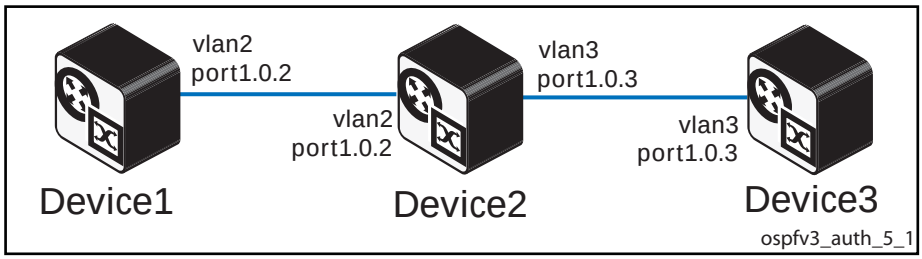
OSPFv3 Encryption with Two VLANs

This example shows how to configure three devices with OSPFv3 encryption on a per VLAN basis. Note that to increase security, rather than using a single SPI for the entire area, this example uses a different security profile for each VLAN in the topology shown.

See section **Configuring OSPFv3 Encryption on a VLAN** for the sequence of console command entries. For detailed command information, see the **ipv6 ospf encryption spi esp** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-16: OSPFv3 Encryption with two VLANs topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2

 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan2
 description Device1_to_Device2
 ipv6 address 2001:db8:2::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf encryption ipsec spi 1000 esp 3des
1234567890abcdef1234567890abcdef12
!
ipv6 forwarding
ipv6 ospf display route single-line
!
router ipv6 ospf process1
 router-id 192.168.1.1
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2

vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!!
interface vlan2
 description Device2_to_Device1
 ipv6 address 2001:db8:2::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf encryption ipsec spi 1000 esp 3des
1234567890abcdef1234567890abcdef12
!
interface vlan3
 description Device2_to_Device3
 ipv6 address 2001:db8:3::1/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf encryption ipsec spi 1200 esp 3des
7890abcdef1234567890abcdef12123456
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
```

Device3 configuration

See the below configuration for a device with the hostname **Device3**:

```
hostname Device3
!
vlan database
 vlan 3 state enable
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan3
 description Device3_to_Device2
 ipv6 address 2001:db8:3::2/64
 ipv6 enable
 ipv6 router ospf area 0 tag process1
 ipv6 ospf encryption ipsec spi 1200 esp 3des
7890abcdef1234567890abcdef12123456
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 3.3.3.3
```

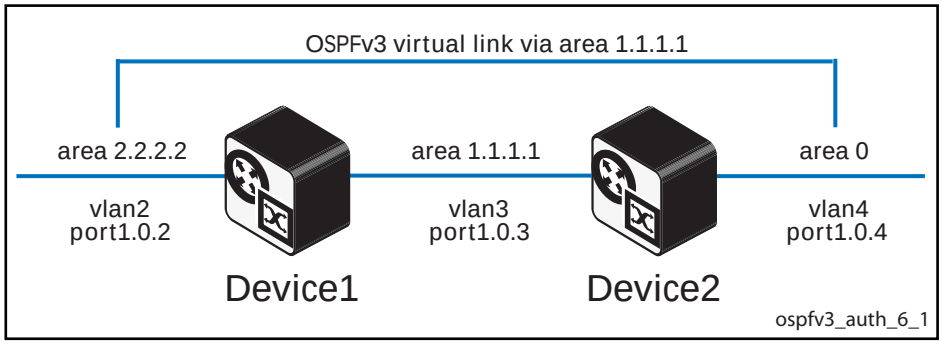
OSPFv3 Authentication for a Virtual Link

In this example, there is an OSPFv3 virtual link configured from area 2.2.2.2 to the OSPFv3 backbone area 0, via transit area 1.1.1.1, with OSPFv3 authentication configured.

See section **Configuring OSPFv3 Authentication and Encryption for a Virtual Link** for the sequence of console command entries. For detailed command information, see the **area virtual-link authentication ipsec spi** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-17: OSPFv3 Authentication for a Virtual Link topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan2
 ipv6 address 2001:db8:2::1/64
 ipv6 router ospf area 2.2.2.2 tag process1
!
interface vlan3
 description Device1_to_Device2
 ipv6 address 2001:db8:3::1/64
 ipv6 router ospf area 1.1.1.1 tag process1
 ipv6 ospf authentication ipsec spi 256 md5
1234567890abcdef1234567809abcdef
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.1
 area 1.1.1.1 virtual-link 192.168.1.2
 area 1.1.1.1 virtual-link 192.168.1.2 authentication ipsec spi
256 md5 1234567890abcdef1234567809abcdef
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2
!
vlan database
 vlan 3,4 state enable
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface vlan3
 description Device2_to_Device1
 ipv6 address 2001:db8:3::2/64
 ipv6 router ospf area 1.1.1.1 tag process1
 ipv6 ospf authentication ipsec spi 256 md5
1234567890abcdef1234567809abcdef
!
interface vlan4
 ipv6 address 2001:db8:4::1/64
 ipv6 router ospf area 0 tag process1
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
 area 1.1.1.1 virtual-link 192.168.1.1
 area 1.1.1.1 virtual-link 192.168.1.2 authentication ipsec spi
256 md5 1234567890abcdef1234567809abcdef
```

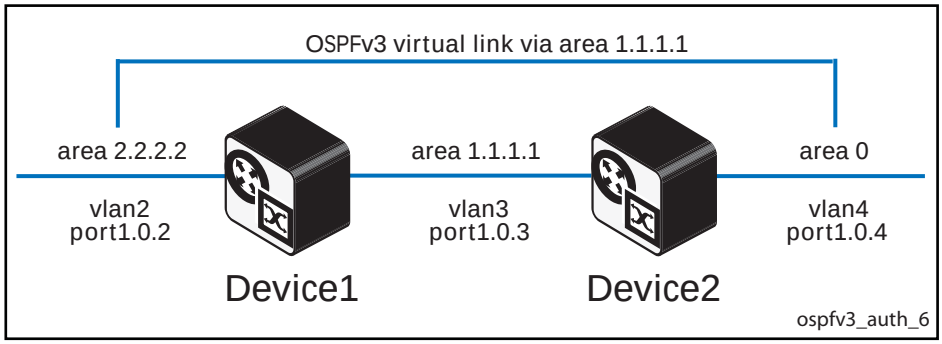
OSPFv3 Encryption for a Virtual Link

In this example, there is an OSPFv3 virtual link configured from OSPFv3 area 2.2.2.2 to the OSPFv3 backbone area 0, via transit area 1.1.1.1, with OSPFv3 encryption configured.

See section **Configuring OSPFv3 Authentication and Encryption for a Virtual Link** for the sequence of console command entries. For detailed command information, see the **area virtual-link encryption ipsec spi** command.

Note that **bold** command entries in configuration output show hostnames and interfaces.

Figure 43-18: OSPFv3 Encryption for a Virtual Link topology:



Device1 configuration

See the below configuration for a device with the hostname **Device1**:

```
hostname Device1
!
vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan2
 ipv6 address 2001:db8:2::1/64
 ipv6 router ospf area 2.2.2.2 tag process1
!
interface vlan3
 description Device1_to_Device2
 ipv6 address 2001:db8:3::1/64
 ipv6 router ospf area 1.1.1.1 tag process1
 ipv6 ospf encryption ipsec spi 1000 esp 3des
1234567890abcdef1234567890abcdef1234567890abcdef md5
1234567890abcdef1234567890abcdef
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.1
 area 1.1.1.1 virtual-link 192.168.1.2
 area 1.1.1.1 virtual-link 192.168.1.2 encryption ipsec spi 2000
esp 3des fedcba0987654321fedcba0987654321fedcba0987654321 md5
fedcba0987654321fedcba0987654321
```

Device2 configuration

See the below configuration for a device with the hostname **Device2**:

```
hostname Device2
!
vlan database
 vlan 3,4 state enable
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface vlan3
 description Device2_to_Device1
 ipv6 address 2001:db8:3::2/64
 ipv6 router ospf area 1.1.1.1 tag process1
 ipv6 ospf encryption ipsec spi 1000 esp 3des
1234567890abcdef1234567890abcdef1234567890abcdef md5
1234567890abcdef1234567890abcdef
!
interface vlan4
 ipv6 address 2001:db8:4::1/64
 ipv6 router ospf area 0 tag process1
!
ipv6 forwarding
!
router ipv6 ospf process1
 router-id 192.168.1.2
 area 1.1.1.1 virtual-link 192.168.1.1
 area 1.1.1.1 virtual-link 192.168.1.1 encryption ipsec spi 2000
esp 3des fedcba0987654321fedcba0987654321fedcba0987654321 md5
fedcba0987654321fedcba0987654321
```

Chapter 44: OSPFv3 for IPv6 Commands

Command List 44.3
abr-type 44.3
area authentication ipsec spi 44.4
area default-cost (IPv6 OSPF) 44.6
area encryption ipsec spi esp 44.7
area range (IPv6 OSPF) 44.10
area stub (IPv6 OSPF) 44.11
area virtual-link (IPv6 OSPF) 44.12
area virtual-link authentication ipsec spi 44.14
area virtual-link encryption ipsec spi 44.16
auto-cost reference bandwidth (IPv6 OSPF) 44.19
bandwidth (duplicate) 44.21
clear ipv6 ospf process 44.22
debug ipv6 ospf events 44.22
debug ipv6 ospf ifsm 44.23
debug ipv6 ospf lsa 44.24
debug ipv6 ospf nfsm 44.25
debug ipv6 ospf packet 44.26
debug ipv6 ospf route 44.27
default-metric (IPv6 OSPF) 44.28
distance (IPv6 OSPF) 44.29
distribute-list (IPv6 OSPF) 44.31
ipv6 ospf authentication spi 44.33
ipv6 ospf cost 44.35
ipv6 ospf dead-interval 44.36
ipv6 ospf display route single-line 44.37
ipv6 ospf encryption spi esp 44.38
ipv6 ospf hello-interval 44.41
ipv6 ospf network 44.42
ipv6 ospf priority 44.43
ipv6 ospf retransmit-interval 44.44
ipv6 ospf transmit-delay 44.45
ipv6 router ospf area 44.46
max-concurrent-dd (IPv6 OSPF) 44.47
passive-interface (IPv6 OSPF) 44.48
redistribute (IPv6 OSPF) 44.49
restart ipv6 ospf graceful 44.50
router ipv6 ospf 44.51
router-id (IPv6 OSPF) 44.52
show debugging ipv6 ospf 44.53
show ipv6 ospf 44.54
show ipv6 ospf database 44.55
show ipv6 ospf database external 44.57
show ipv6 ospf database grace 44.59
show ipv6 ospf database inter-prefix 44.61
show ipv6 ospf database inter-router 44.63
show ipv6 ospf database intra-prefix 44.65

show ipv6 ospf database link 44.67
show ipv6 ospf database network 44.69
show ipv6 ospf database router 44.70
show ipv6 ospf interface 44.74
show ipv6 ospf neighbor 44.76
show ipv6 ospf route 44.77
show ipv6 ospf virtual-links 44.78
summary-address (IPv6 OSPF) 44.79
timers spf (IPv6 OSPF) (deprecated) 44.81
timers spf exp (IPv6 OSPF) 44.82
undebug ipv6 ospf events 44.83
undebug ipv6 ospf ifsm 44.83
undebug ipv6 ospf lsa 44.83
undebug ipv6 ospf nfsm 44.83
undebug ipv6 ospf packet 44.83
undebug ipv6 ospf route 44.83

Command List

This chapter provides an alphabetical reference of commands used to configure OSPFv3 for IPv6. For more information on this topic, see **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration**.

abr-type

Use this command to set an OSPF Area Border Router (ABR) type.

Use the **no** variant of this command to revert the ABR type to the default setting (Cisco).

Syntax

```
abr-type {cisco|ibm|standard}
no abr-type {cisco|ibm|standard}
```

Parameter	Description
`cisco`	Specifies an alternative ABR using Cisco implementation (RFC 3509). This is the default ABR type.
`ibm`	Specifies an alternative ABR using IBM implementation (RFC 3509).
`standard`	Specifies a standard behavior ABR (RFC 2328).

Default ABR type `cisco`

Mode Router Configuration

Usage Specifying the ABR type allows better interoperation between different implementations. This command is specially useful in a multi-vendor environment. The different ABR types are:

- Cisco ABR Type: By this definition, a router is considered an ABR if it has more than one area actively attached and one of them is the backbone area.
- IBM ABR Type: By this definition, a router is considered an ABR if it has more than one area actively attached and the backbone area is configured. In this case the configured backbone need not be actively connected.
- Standard ABR Type: By this definition, a router is considered an ABR if it has more than one area actively attached to it.

Example

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# abr-type ibm
```

area authentication ipsec spi

Use this command in Router Configuration mode to enable either MD5 (Message-Digest 5) or SHA1 (Secure Hash Algorithm 1) authentication for a specified OSPF area.

Use the **no** variant of this command in Router Configuration mode to disable the authentication configured for a specified OSPF area.

Syntax

```
area <area-id> authentication ipsec spi <256-4294967295>
    {md5 <MD5-key>|sha1 <SHA1-key>}

no area <area-id> authentication ipsec spi <256-4294967295>
```

Parameter	Description	
<area-id>	The OSPF area that you are specifying the summary route default-cost for. This can be entered in either dotted decimal format or normal decimal format. Use one of the following formats:	
	<ip-addr>	OSPF area-ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example, the values 0.0.1.2 and decimal 258 would both define the same area-ID.	
<256-4294967295>	Specify an SPI (Security Parameters Index) value in the range 256 to 4294967295, entered as a decimal integer.	
md5	Specify the MD5 (Message-Digest 5) hashing algorithm.	
<MD5-key>	Enter an MD5 key containing up to 32 hexadecimal characters.	
sha1	Specify the SHA-1 (Secure Hash Algorithm 1) hashing algorithm.	
<SHA1-key>	Enter an SHA-1 key containing up to 40 hexadecimal characters.	

Mode Router Configuration

Usage Use this command on an OSPFv3 area, use the **area virtual-link authentication ipsec spi** command on an OSPFv3 area virtual link. Configure the same SPI (Security Parameters Index) value on all interfaces that connect to the same link. SPI values are used by link interfaces. Use a different SPI value for a different link interface when using OSPFv3 with link interfaces.

Use the **sha1** keyword to choose SHA-1 authentication instead of entering the **md5** keyword to use MD5 authentication. The SHA-1 algorithm is more secure than the MD5 algorithm. SHA-1 uses a 40 hexadecimal character key instead of a 32 hexadecimal character key as used for MD5 authentication.

See the section **OSPFv3 Authentication and Encryption Overview** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for configuration examples and topologies.

Note

You can configure an authentication security policy (SPI) on an OSPFv3 area with this command, or on a VLAN interface with the **ipv6 ospf authentication spi** command.

When you configure authentication for an area, the security policy is applied to all VLAN interfaces in the area. However, Allied Telesis recommends a different authentication security policy is applied to each interface for higher security.

If you apply the `ipv6 ospf authentication null` command this affects authentication configured on both the VLAN interface and the OSPFv3 area.

This is due to OSPFv3 hello messages ingressing VLAN interfaces, which are part of area authentication, not being authenticated. So neighbors time out.

Example To enable MD5 authentication with a 32 hexadecimal character key for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 authentication ipsec spi 1000 md5
1234567890ABCDEF1234567890ABCDEF
```

To enable SHA-1 authentication with a 40 hexadecimal character key for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 authentication ipsec spi 1000 sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To disable authentication for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no area 1 authentication ipsec spi 1000
```

Related Commands

area encryption ipsec spi esp
area virtual-link authentication ipsec spi
area virtual-link encryption ipsec spi
ipv6 ospf authentication spi
ipv6 ospf encryption spi esp
show ipv6 ospf

area default-cost (IPv6 OSPF)

This command specifies a cost for the default summary route sent into a stub area.

The **no** variant of this command removes the assigned default-route cost.

Syntax

```
area <area-id> default-cost <0-16777215>
no area <area-id> default-cost
```

Parameter	Description	
<area-id>	The OSPF area that you are specifying the summary route default-cost for. This can be entered in either dotted decimal format or normal decimal format. Use one of the following formats:	
	<ip-addr>	OSPF area-ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example, the values 0.0.1.2 and decimal 258 would both define the same area-ID.	
`default-cost`	Indicates the cost for the default summary route used for a stub area. Default: **1**	

Mode Router Configuration

Usage The default-cost option provides the metric for the summary default route, generated by the area border router, into the stub area. Use this option only on an area border router that is attached to the stub area.

Example To set the default cost to 10 in area 1 for the OSPF process P2, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf P2
awplus(config-router)# area 1 default-cost 10
```

Related Commands area stub (IPv6 OSPF)

area encryption ipsec spi esp

Use this command in Router Configuration mode to enable either AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) or 3DES (Triple Data Encryption Standard) ESP (Encapsulating Security Payload) encryption for a specified OSPF area.

Use the **no** variant of this command in Router Configuration mode to disable the encryption configured for a specified OSPF area.

Syntax

```
area <area-id> encryption ipsec spi <256-4294967295> esp
   {aes-cbc <AES-CBC-key>|3des <3DES-key>|null}{md5 <MD5-key>|
   sha1 <SHA1-key>}

no area <area-id> encryption ipsec spi <256-4294967295>
```

Parameter	Description	
<area-id>	The OSPF area that you are specifying the summary route default-cost for. This can be entered in either dotted decimal format or normal decimal format. Use one of the following formats:	
	<ip-addr>	OSPF area-ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example, the values 0.0.1.2 and decimal 258 would both define the same area-ID.	
<256-4294967295>	Specify an SPI (Security Parameters Index) value in the range 256 to 4294967295, entered as a decimal integer.	
`esp`	Specify the esp keyword (Encapsulating Security Payload) to then apply either AES-CBC or 3DES encryption.	
`aes-cbc`	Specify this keyword to enable AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) encryption.	
<AES-CBC-key>	Enter an AES-CBC key containing either 32, 48, or 64 hexadecimal characters.	
`3des`	Specify 3DES (Triple Data Encryption Standard) encryption.	
<3DES-key>	Enter a 3DES key containing 48 hexadecimal characters.	
`null`	Specify ESP without AES-CBC or 3DES encryption applied.	
`md5`	Specify the MD5 (Message-Digest 5) encryption algorithm.	
<MD5-key>	Enter an MD5 key containing 32 hexadecimal characters.	
`sha1`	Specify the SHA-1 (Secure Hash Algorithm 1) encryption algorithm.	
<SHA1-key>	Enter an SHA-1 key containing 40 hexadecimal characters.	

Mode Router Configuration

Usage When you issue this command, authentication and encryption are both enabled.

Use this command on an OSPFv3 area, use the **area virtual-link encryption ipsec spi** command on an OSPFv3 area virtual link. Configure the same SPI (Security Parameters Index) value on all interfaces that connect to the same link. SPI values are used by link interfaces. Use a different SPI value for a different link interface when using OSPFv3 with link interfaces.

Security is achieved using the IPv6 ESP extension header. The IPv6 ESP extension header is used to provide confidentiality, integrity, authentication, and confidentiality. Authentication fields are removed from OSPF for IPv6 packet headers, so applying IPv6 ESP extension headers are required for integrity, authentication, and confidentiality.

Use the **sha1** keyword to choose SHA-1 authentication instead of entering the **md5** keyword to use MD5 authentication. The SHA-1 algorithm is more secure than the MD5 algorithm. SHA-1 uses a 40 hexadecimal character key instead of a 32 hexadecimal character key as used for MD5 authentication.

See the section **OSPFv3 Authentication and Encryption Overview** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for configuration examples and topologies.

Note

You can configure an encryption security policy (SPI) on an OSPFv3 area with this command, or on a VLAN interface with the **ipv6 ospf encryption spi esp** command.

When you configure encryption for an area, the security policy is applied to all VLAN interfaces in the area. However, Allied Telesis recommends a different encryption security policy is applied to each interface for higher security.

If you apply the `ipv6 ospf encryption null` command this affects encryption configured on both the VLAN interface and the OSPFv3 area.

This is due to OSPFv3 hello messages ingressing VLAN interfaces, which are part of area encryption, not being being encrypted. So neighbors time out.

Example To enable ESP encryption, but not apply an AES-CBC key or an 3DES key, and MD5 authentication with a 32 hexadecimal character key for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 encryption ipsec spi 1000 esp null
md5 1234567890ABCDEF1234567890ABCDEF
```

To enable ESP encryption, but not apply an AES-CBC key or an 3DES key, and SHA-1 authentication with a 40 hexadecimal character key for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 encryption ipsec spi 1000 esp null
sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To enable ESP encryption with a 48 hexadecimal character 3DES key and a 32 hexadecimal character MD5 authentication for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 encryption ipsec spi 1000 esp 3des
1234567890ABCDEF1234567890ABCDEF12345678
90ABCDEF md5
1234567890ABCDEF1234567890ABCDEF
```

To enable ESP encryption with a 32 hexadecimal character AES-CBC key, and a 40 hexadecimal character SHA-1 authentication key for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 encryption ipsec spi 1000 esp aes-
cbc 1234567890ABCDEF1234567890ABCDEF
sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To disable ESP encryption for OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no area 1 encryption ipsec spi 1000
```

Related Commands

area authentication ipsec spi
area virtual-link authentication ipsec spi
area virtual-link encryption ipsec spi
ipv6 ospf authentication spi
ipv6 ospf encryption spi esp
show ipv6 ospf

area range (IPv6 OSPF)

Use this command to summarize OSPFv3 routes at an area boundary, configuring an IPv6 address range which consolidates OSPFv3 routes. By default, this feature is not enabled.

A summary route created by this command is then advertised to other areas by the Area Border Routers (ABRs). In this way, routing information is condensed at area boundaries and outside the area so that routes are exchanged between areas in an efficient manner.

If the network numbers in an area are arranged into sets of contiguous routes, the ABRs can be configured to advertise a summary route that covers all the individual networks within the area that fall into the specified range.

The **no** variant of this command disables this function and restores default behavior.

Syntax

```
area <area-id> range <ipv6address/prefix-length> [advertise|not-
    advertise]

no area <area-id> range <ipv6address/prefix-length>
```

Parameter	Description	
<area-id>	The OSPFv3 area that you summarizing the routes for. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format.	
	<A.B.C.D>	OSPF area-ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example the values 0.0.1.2 and decimal 258 would both define the same area-ID.	
<ip-addr/ prefix-length>	The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.	
`advertise`	Advertise this range as a summary route into other areas.	
`not-advertise`	Do not advertise this range.	

Default The area range is not configured by default. The area range is advertised if it is configured.

Mode Router Configuration

Usage You can configure multiple ranges on a single area with multiple instances of this command, so OSPFv3 summarizes addresses for different sets of IPv6 address ranges.

Ensure OSPFv3 IPv6 routes exist in the area range for advertisement before using this command.

Example

```
awplus# configure terminal
awplus(config)# router ipv6 ospf P2
awplus(config-router)# area 1 range 2000::/3
```

area stub (IPv6 OSPF)

This command defines an OSPF area as a stub area. By default, no stub area is defined.

Use this command when routers in the area do not require learning about external LSAs. You can define the area as a totally stubby area by configuring the Area Border Router of that area using the **area stub no-summary** command.

The **no** variant of this command removes this definition.

Syntax

```
area <area-id> stub [no-summary]
no area <area-id> stub [no-summary]
```

Parameter	Description	
<area-id>	The OSPF area that you are configuring as a stub area. Use one of the following formats: This can be entered in either dotted decimal format or normal decimal format. For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area-ID.	
	<A.B.C.D>	OSPF area-ID, expressed in the IPv4 address format *<A.B.C.D>*.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area-ID.	
no-summary	Stops an ABR from sending summary link advertisements into the stub area.	

Mode Router Configuration

Usage There are two stub area router configuration commands: the **area stub** and **area default-cost** commands. In all routers attached to the stub area, configure the area by using the **area stub** command. For an area border router (ABR) attached to the stub area, also use the **area default-cost** command.

Example

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# area 100 stub
```

Related Commands area default-cost (IPv6 OSPF)

area virtual-link (IPv6 OSPF)

This command configures a link between a non-backbone area and the backbone, through other non-backbone areas.

In OSPF, all non-backbone areas must be connected to a backbone area. If the connection to the backbone is lost, the virtual link repairs the connection.

The **no** variant of this command removes the virtual link.

Syntax

```
area <area-id> virtual-link <router-id>

no area <area-id>  virtual-link <router-id>

area <area-id>  virtual-link <router-id>

no area <area-id>  virtual-link <router-id>

area <area-id>  virtual-link <router-id> [hello-interval <1-65535>]
   [retransmit-interval <1-65535>] [transmit-delay <1-65535>]

no area <area-id>  virtual-link <router-id> [hello-interval]
   [retransmit-interval] [transmit-delay]
```

Parameter	Description	
<area-id>	The area-ID of the transit area that the virtual link passes through. This can be entered in either dotted decimal format or normal decimal format as shown below.	
	<A.B.C.D>	OSPF area-ID, expressed in the IPv4 address format *<A.B.C.D>*.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example the values dotted decimal 0.0.1.2 and decimal 258 would both define the same area-ID.	
<router-id>	The OSPF router ID of the virtual link neighbor.	
dead-interval	If no packets are received from a particular neighbor for dead-interval seconds, the router considers the neighbor router to be off-line. Default: 40 seconds	
	<1-65535>	The number of seconds in the interval.
hello-interval	The interval the router waits before it sends a hello packet. Default: 10 seconds	
	<1-65535>	The number of seconds in the interval.
retransmit-interval	The interval the router waits before it retransmits a packet. Default: 5 seconds	
	<1-65535>	The number of seconds in the interval.
transmit-delay	The interval the router waits before it transmits a packet. Default: 1 seconds	
	<1-65535>	The number of seconds in the interval.

Mode Router Configuration

Usage

You can configure virtual links between any two backbone routers that have an interface to a common non-backbone area. The protocol treats these two routers, joined by a virtual link, as if they were connected by an unnumbered point-to-point network. To configure a virtual link, you require:

- The transit area-ID, i.e. the area-ID of the non-backbone area that the two backbone routers are both connected to.
- The corresponding virtual link neighbor's router ID. To see the router ID use the **show ipv6 ospf** command.

Configure the **hello-interval** to be the same for all routers attached to a common network. A short **hello-interval** results in the router detecting topological changes faster but also an increase in the routing traffic.

The **retransmit-interval** is the expected round-trip delay between any two routers in a network. Set the value to be greater than the expected round-trip delay to avoid needless retransmissions.

The **transmit-delay** is the time taken to transmit a link state update packet on the interface. Before transmission, the link state advertisements in the update packet, are incremented by this amount. Set the **transmit-delay** to be greater than zero. Also, take into account the transmission and propagation delays for the interface.

Example

To configure a virtual link through area 1to the router with router-ID 10.10.11.50, use the following commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# area 1 virtual-link 10.10.11.50 hello 5
dead 10
```

Related Commands

show ipv6 ospf

area virtual-link authentication ipsec spi

Use this command in Router Configuration mode to enable authentication for virtual links in a specified OSPF area.

Use the **no** variant of this command in Router Configuration mode to disable authentication for virtual links in a specified OSPF area.

Syntax

```
area <area-id> virtual-link <router-ID> authentication ipsec
    spi <256-4294967295> {md5 <MD5-key>|sha1 <SHA1-key>}

no area <area-id> virtual-link <router-ID> authentication ipsec
    spi <256-4294967295>
```

Parameter	Description	
<area-id>	The OSPF area that you are specifying the summary route default-cost for. This can be entered in either dotted decimal format or normal decimal format. Use one of the following formats:	
	<ip-addr>	OSPF area-ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example, the values 0.0.1.2 and decimal 258 would both define the same area-ID.	
`virtual-link`	Specify a virtual link and its parameters.	
<router-ID>	Enter a router ID associated with a virtual link neighbor in IPv4 address format A.B.C.D.	
`authentication`	Specify this keyword to enable authentication.	
`ipsec`	Specify this keyword to use IPsec authentication.	
`spi`	Specify this keyword to set the SPI (Security Parameters Index).	
<256-4294967295>	Specify an SPI (Security Parameters Index) value in the range 256 to 4294967295, entered as a decimal integer.	
`md5`	Specify the MD5 (Message-Digest 5) encryption algorithm.	
<MD5-key>	Enter an MD5 key containing 32 hexadecimal characters.	
`sha1`	Specify the SHA-1 (Secure Hash Algorithm 1) encryption algorithm.	
<SHA1-key>	Enter an SHA-1 key containing 40 hexadecimal characters.	

Mode Router Configuration

Usage Use this command on an OSPFv3 area virtual link, use the **area authentication ipsec spi** command on an OSPFv3 area. Configure the same SPI (Security Parameters Index) value on all interfaces that connect to the same link. SPI values are used by link interfaces. Use a different SPI value for a different link interface when using OSPFv3 with link interfaces.

OSPFv3 areas are connected to a backbone area. Virtual links can be configured to repair lost connections to a backbone area for OSPFv3 areas. To configure an OSPFv3 virtual link, use a router ID instead of the IPv6 prefix of the router.

Use the **sha1** keyword to choose SHA-1 authentication instead of entering the **md5** keyword to use MD5 authentication. The SHA-1 algorithm is more secure than the MD5 algorithm. SHA-1 uses a 40 hexadecimal character key instead of a 32 hexadecimal character key as used for MD5 authentication.

See the section **OSPFv3 Virtual Links** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption for a Virtual Link** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for configuration examples and topologies.

Example To enable MD5 authentication with a 32 hexadecimal character key for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 virtual-link 10.0.0.1
                       authentication ipsec spi 1000 md5
                       1234567890ABCDEF1234567890ABCDEF
```

To enable SHA-1 authentication with a 40 hexadecimal character key for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 virtual-link 10.0.0.1
                       authentication ipsec spi 1000 sha1
                       1234567890ABCDEF1234567890ABCDEF12345678
```

To disable authentication for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no area 1 virtual-link ipsec spi 1000
```

Related Commands
area authentication ipsec spi
area encryption ipsec spi esp
area virtual-link encryption ipsec spi
show ipv6 ospf virtual-links

area virtual-link encryption ipsec spi

Use this command in Router Configuration mode to enable either AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) or 3DES (Triple Data Encryption Standard) ESP (Encapsulating Security Payload) encryption for virtual links in a specified OSPF area.

Use the **no** variant of this command in Router Configuration mode to disable encryption configured for virtual links in a specified OSPF area.

Syntax

```
area <area-id> virtual-link <router-ID> encryption ipsec spi
   <256-4294967295> esp {aes-cbc <AES-CBC-key>|3des <3DES-key>|
   null}{md5 <MD5-key>|sha1 <SHA1-key>}

no area <area-id> encryption ipsec spi <256-4294967295>
```

Parameter	Description	
<area-id>	The OSPF area that you are specifying the summary route default-cost for. This can be entered in either dotted decimal format or normal decimal format. Use one of the following formats:	
	<ip-addr>	OSPF area-ID expressed in IPv4 address format A.B.C.D.
	<0-4294967295>	OSPF area-ID expressed as a decimal number within the range shown.
	For example, the values 0.0.1.2 and decimal 258 would both define the same area-ID.	
`virtual-link`	Specify a virtual link and its parameters.	
<router-ID>	Enter a router ID associated with a virtual link neighbor in IPv4 address format A.B.C.D.	
`encryption`	Specify this keyword to enable encryption.	
`ipsec`	Specify this keyword to use IPsec authentication.	
`spi`	Specify this keyword to set the SPI (Security Parameters Index).	
<256-4294967295>	Specify an SPI (Security Parameters Index) value in the range 256 to 4294967295, entered as a decimal integer.	
`esp`	Specify the esp keyword (Encapsulating Security Payload) to then apply either AES-CBC or 3DES encryption.	
`aes-cbc`	Specify this keyword to enable AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) encryption.	
<AES-CBC-key>	Enter an AES-CBC key containing either 32, 48, or 64 hexadecimal characters.	
`3des`	Specify 3DES (Triple Data Encryption Standard) encryption.	
<3DES-key>	Enter a 3DES key containing 48 hexadecimal characters.	
`null`	Specify ESP without AES-CBC or 3DES encryption applied.	
`md5`	Specify the MD5 (Message-Digest 5) encryption algorithm.	
<MD5-key>	Enter an MD5 key containing 32 hexadecimal characters.	
`sha1`	Specify the SHA-1 (Secure Hash Algorithm 1) encryption algorithm.	
<SHA1-key>	Enter an SHA-1 key containing 40 hexadecimal characters.	

Mode Router Configuration

Usage When you issue this command, authentication and encryption are both enabled.

Use this command on an OSPFv3 area virtual link, use the **area encryption ipsec spi esp** command on an OSPFv3 area. Configure the same SPI (Security Parameters Index) value on all interfaces that connect to the same link. SPI values are used by link interfaces. Use a different SPI value for a different link interface when using OSPFv3 with link interfaces.

Security is achieved using the IPv6 ESP extension header. ESP is used to provide confidentiality, integrity, authentication, and confidentiality. Authentication fields are removed from OSPF for IPv6 packet headers. The IPv6 ESP extension header is required for integrity, authentication, and confidentiality.

Note that interface configuration takes priority over area configuration. If an interface configuration is removed then an area configuration is applied to an interface instead.

Use the **sha1** keyword to choose SHA-1 authentication instead of entering the **md5** keyword to use MD5 authentication. The SHA-1 algorithm is more secure than the MD5 algorithm. SHA-1 uses a 40 hexadecimal character key instead of a 32 hexadecimal character key as used for MD5 authentication.

See the section **OSPFv3 Virtual Links** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption for a Virtual Link** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for configuration examples and topologies.

Example To enable ESP encryption, but not apply an AES-CBC key or a 3DES key, and MD5 authentication with a 32 hexadecimal character key for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 virtual-link 10.0.0.1 encryption
ipsec spi 1000 esp null md5
1234567890ABCDEF1234567890ABCDEF
```

To enable ESP encryption, but not apply an AES-CBC key or a 3DES key, and SHA-1 authentication with a 40 hexadecimal character key for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 virtual-link 10.0.0.1 encryption
ipsec spi 1000 esp null sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To enable ESP encryption with a 32 hexadecimal character AES-CBC key and a 40 hexadecimal character SHA-1 authentication key for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 virtual-link 10.0.0.1 encryption
ipsec spi 1000 esp aes-cbc
1234567890ABCDEF1234567890ABCDEF sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To enable ESP encryption with a 48 hexadecimal character 3DES key and a 40 hexadecimal character SHA-1 authentication key for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# area 1 virtual-link 10.0.0.1 encryption
ipsec spi 1000 esp 3des
1234567890ABCDEF1234567890ABCDEF12345678
90ABCDEF sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To disable authentication for virtual links in OPSPF area 1, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no area 1 virtual-link 10.0.0.1
authentication ipsec spi 1000
```

Related Commands

area authentication ipsec spi
area encryption ipsec spi esp
area virtual-link authentication ipsec spi
show ipv6 ospf virtual-links

auto-cost reference bandwidth (IPv6 OSPF)

This command controls how OSPF calculates default metrics for the interface.

Use the **no** variant of this command to assign cost based only on the interface bandwidth.

Syntax

```
auto-cost reference-bandwidth <1-4294967>
no auto-cost reference-bandwidth
```

Parameter	Description
<1-4294967>	The reference bandwidth, measured in Mbits per second (Mbps).

Default 1000 Mbps

Usage By default, OSPF calculates the OSPF metric for an interface by dividing the reference bandwidth by the interface bandwidth. The default for the reference bandwidth is 1000 Mbps. As a result, if this default is used, there is very little difference between the metrics applied to interfaces of increasing bandwidth beyond 1000 Mbps.

The auto-cost command is used to alter this reference bandwidth in order to give a real difference between the metrics of high bandwidth links of differing bandwidths. In a network that has multiple links with high bandwidths, specify a larger reference bandwidth value to differentiate the costs on those links.

Cost is calculated by dividing the reference bandwidth (Mbps) by the layer 3 interface (Switched Virtual Interface (SVI), Loopback or Ethernet interface) bandwidth. Interface bandwidth may be altered by using the **bandwidth (duplicate)** command on page 44.21 as the SVI does not auto detect the bandwidth based on the speed of associated switch ports.

When the reference bandwidth calculation results in a cost integer greater than 1 but contains a fractional value (value after the decimal point), the result rounds down to the nearest integer. The following example shows how the cost is calculated.

The reference bandwidth is 1000 Mbps and the interface bandwidth is 7 Mbps.

Calculation = 1000/7

Calculation result = 142.85 (integer of 142, fractional value of 0.85)

Result after rounding down to the nearest integer = 142 (Interface cost is 142)

When the reference bandwidth calculation results in a cost less than 1, it is rounded up to the nearest integer which is 1. The following example shows how the cost is calculated.

The reference bandwidth is 1000 Mbps and the interface bandwidth is 10000 Mbps.

Calculation = 1000/10000

Calculation result = 0.1

Result after rounding up to the nearest integer = 1 (Interface cost is 1)

The auto-cost reference bandwidth value should be consistent across all OSPF routers in the OSPF process.

Note that using the **ipv6 ospf cost** command on page 44.35 on a layer 3 interface will override the cost calculated by the reference bandwidth command.

Mode Router Configuration

Example

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 20
awplus(config-router)# auto-cost reference-bandwidth 1000
```

Related Commands **ipv6 ospf cost**

bandwidth (duplicate)

Use this command to specify the maximum bandwidth to be used for each VLAN interface.

The bandwidth value is in bits. OSPF uses this to calculate metrics for the VLAN interface.

The **no** variant of this command removes any applied bandwidth value and replaces it with a value equal to the lowest port speed within that VLAN.

Syntax

```
bandwidth  <bandwidth-setting>
no bandwidth
```

Parameter	Description
<bandwidth-setting>	Sets the bandwidth for the interface. Enter a value in the range 1 to 10000000000 bits per second. Note that to avoid entering many zeros, you can add k, m, or g to internally add 3, 6 or 9 zeros to the number entered. For example entering 1k is the same as entering 1000.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# bandwidth 1000000
```

Related Commands

show running-config
show running-config access-list
show interface

clear ipv6 ospf process

This command clears and restarts the IPv6 OSPF routing process. Specify the Process ID to clear one particular OSPF process. When no Process ID is specified, this command clears all running OSPF processes.

Syntax `clear ipv6 ospf [<0-65535>] process`

Parameter	Description
`<0-65535>`	The routing process ID.

Mode Privileged Exec

Example

```
awplus# clear ipv6 ospf process
```

debug ipv6 ospf events

This command enables IPv6 OSPF debugging for event troubleshooting.

To enable all debugging options, specify **debug ipv6 ospf event** with no additional parameters.

The **no** and **undebug** variants of this command disable OSPF debugging. Using this command with no parameters entered, will disable debugging for all parameter options.

Syntax `debug ipv6 ospf events [abr] [asbr] [os][router] [vlink]`

`no debug ipv6 ospf events [abr] [asbr] [os] [router] [vlink]`

Parameter	Description
`abr`	Shows ABR events.
`asbr`	Shows ASBR events.
`router`	Shows other router events.
`os`	Shows OS events.
`vlink`	Shows virtual link events.

Mode Privileged Exec and Global Configuration

Example To enable IPv6 event debugging and show ABR events, use the following command:

```
awplus# debug ipv6 ospf events asbr
```

debug ipv6 ospf ifsm

This command specifies debugging options for IPv6 OSPF Interface Finite State Machine (IFSM) troubleshooting.

To enable all debugging options, specify **debug ipv6 ospf ifsm** with no additional parameters.

The **no** and **undebug** variants of this command disable IPv6 OSPF IFSM debugging. Use these commands without parameters to disable all the options.

Syntax

```
debug ipv6 ospf ifsm [events] [status] [timers]
no debug ipv6 ospf ifsm [events] [status] [timers]
```

Parameter	Description
`events`	Displays IFSM event information.
`status`	Displays IFSM status information.
`timers`	Displays IFSM timer information.

Mode Privileged Exec and Global Configuration

Example To specify IPv6 OSPF debugging options to display IPv6 OSPF IFSM events information, use the following commands:

```
awplus# debug ipv6 ospf ifsm events
```

Related Commands **terminal monitor**
undebug ipv6 ospf ifsm

debug ipv6 ospf lsa

This command enables debugging options for IPv6 OSPF Link State Advertisements (LSA) troubleshooting. This displays information related to internal operations of LSAs.

To enable all debugging options, specify **debug ipv6 ospf lsa** with no additional parameters.

The **no** and **undebug** variants of this command disable IPv6 OSPF LSA debugging. Use this command without parameters to disable all the options.

Syntax

```
debug ipv6 ospf lsa [flooding] [generate] [install] [maxage]
    [refresh]

no debug ipv6 ospf lsa [flooding] [generate] [install] [maxage]
    [refresh]
```

Parameter	Description
`flooding`	Displays LSA flooding.
`generate`	Displays LSA generation.
`install`	Show LSA installation.
`maxage`	Shows maximum age of the LSA in seconds.
`refresh`	Displays LSA refresh.

Mode Privileged Exec and Global Configuration

Examples To enable debugging for IPv6 OSPF refresh LSA, use the following commands:

```
awplus# debug ipv6 ospf lsa refresh
```

Related Commands **terminal monitor**
undebug ipv6 ospf lsa

debug ipv6 ospf nfsm

This command enables debugging options for IPv6 OSPF Neighbor Finite State Machines (NFSMs).

To enable all debugging options, specify **debug ipv6 ospf nfsm** with no additional parameters.

The **no** and **undebug** variants of this command disable IPv6 OSPF NFSM debugging. Use this command without parameters to disable all the options.

Syntax

```
debug ipv6 ospf nfsm [events] [status] [timers]
no debug ipv6 ospf nfsm [events] [status] [timers]
```

Parameter	Description
`events`	Displays NFSM event information.
`status`	Displays NSFM status information.
`timers`	Displays NFSM timer information.

Mode Privileged Exec and Global Configuration

Examples To enable IPv6 debugging option to display timer information, use the following command:

```
awplus# debug ipv6 ospf nfsm timers
```

Related Commands **terminal monitor**
undebug ipv6 ospf nfsm

debug ipv6 ospf packet

This command enables debugging options for IPv6 OSPF packets.

To enable all debugging options, specify **debug ipv6 ospf packet** with no additional parameters.

The **no** and **undebug** variants of this command disable IPv6 OSPF packet debugging. Use this command without parameters to disable all options.

Syntax

```
debug ipv6 ospf packet [dd] [detail] [hello] [ls-ack] [ls-request]
    [ls-update] [recv] [send]

no debug ipv6 ospf packet [dd] [detail] [hello] [ls-ack] [ls-request]
    [ls-update] [recv] [send]
```

Parameter	Description
`dd`	Specifies debugging for IPv6 OSPF database descriptions.
`detail`	Sets the debug option to detailed information.
`hello`	Specifies debugging for IPv6 OSPF hello packets.
`ls-ack`	Specifies debugging for IPv6 OSPF link state acknowledgments.
`ls-request`	Specifies debugging for IPv6 OSPF link state requests.
`ls-update`	Specifies debugging for IPv6 OSPF link state updates.
`recv`	Specifies the debug option set for received packets.
`send`	Specifies the debug option set for sent packets.

Mode Privileged Exec and Global Configuration

Examples To enable debugging for hello packets, use the following command:

```
awplus# debug ipv6 ospf packet hello
```

Related Commands

terminal monitor
undebug ipv6 ospf packet

debug ipv6 ospf route

This command enables debugging of route calculation. Use this command without parameters to turn on all the options.

The **no** and **undebug** variants of this command disable IPv6 OSPF route debugging. Use this command without parameters to disable all options.

Syntax

```
debug ipv6 ospf route [ase] [ia] [install] [spf]
no debug ipv6 ospf route [ase] [ia] [install] [spf]
```

Parameter	Description
`ase`	Specifies the debugging of external route calculation.
`ia`	Specifies the debugging of inter-area route calculation.
`install`	Specifies the debugging of route installation.
`spf`	Specifies the debugging of SPF calculation.

Mode Privileged Exec and Global Configuration

Examples To enable IPv6 route debugging of inter-area route calculations, use the following command:

```
awplus# debug ipv6 ospf route ia
```

Related Commands **terminal monitor**
undebug ipv6 ospf route

default-metric (IPv6 OSPF)

This command sets default metric value for routes redistributed into the IPv6 OSPF routing protocol.

The **no** variant of this command returns IPv6 OSPF to using built-in, automatic metric translations, as appropriate for each routing protocol.

Syntax

```
default-metric <0-16777214>
no default-metric [<0-16777214>]
```

Parameter	Description
<1-16777214>	Default metric value appropriate for the specified routing protocol.

Mode Router Configuration

Usage A default metric facilitates redistributing routes even with incompatible metrics. If the metrics do not convert, the default metric provides an alternative and enables the redistribution to continue. The effect of this command is that IPv6 OSPF will use the same metric value for **all** redistributed routes. Use this command in conjunction with the **redistribute (IPv6 OSPF)** command.

Examples

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# default-metric 100

awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# no default-metric
```

Related commands **redistribute (IPv6 OSPF)**

distance (IPv6 OSPF)

This command sets the administrative distance for OSPFv3 routes based on the route type. Your switch uses this value to select between two or more routes to the same destination from two different routing protocols. The route with the smallest administrative distance value is added to the Forwarding Information Base (FIB). See **"Administrative Distance" on page 35.6** for more information.

Use the command **distance ospfv3** to set the distance for an entire category of OSPFv3 routes, rather than the specific routes that pass an access list.

Use the command **distance <1-254>**, with no other parameter, to set the same distance for all OSPFv3 route types.

The **no** variant of this command sets the administrative distance for OSPFv3 routes to the default of 110.

Syntax

```
distance <1-254>

distance ospfv3 {external <1-254>|inter-area <1-254>|
   intra-area <1-254>}

no distance {ospfv3|<1-254>}
```

Parameter	Description
<1-254>	Specify the Administrative Distance value for OSPFv3 routes.
`external`	Sets the distance for routes from other routing domains, learned by redistribution. Specify an OSPFv3 external distance in the range *<1-254>*.
`inter-area`	Sets the distance for all routes from one area to another area. Specify an OSPFv3 inter-area distance in the range *<1-254>*.
`intra-area`	Sets the distance for all routes within an area. Specify an OSPFv3 intra-area distance in the range *<1-254>*.

Default The default OSPFv3 administrative distance is 110. The default Administrative Distance for each type of route (intra, inter, or external) is 110.

Mode Router Configuration

Usage The administrative distance rates the trustworthiness of a routing information source. The distance could be any integer from 0 to 254. A higher distance value indicates a lower trust rating. For example, an administrative distance of 254 indicates that the routing information source cannot be trusted and should be ignored.

Use this command to set the distance for an entire group of routes, rather than a specific route that passes an access list.

Examples To set the following administrative distances for route types in OSPF 100:

- 20 for inter-area routes
- 10 for intra-area routes
- 40 for external routes

use the commands:

```
awplus(config)# router ipv6 ospf 100
awplus(config-router)# distance ospfv3 inter-area 20 intra-area 10
external 40
```

To set the administrative distance for all routes in OSPFv3 100 back to the default of 110, use the commands:

```
awplus(config)# router ipv6 ospf 100
awplus(config-router)# no distance ospfv3
```

distribute-list (IPv6 OSPF)

Use this command in Router Configuration mode to filter incoming or outgoing OSPFv3 route updates from the networks as defined in an associated access-list.

See **Usage** for detailed information about the **in** and **out** distribution list parameters.

The entities that are used to perform filtering are ACLs (Access Control Lists), which match on certain attributes in the routes that are being transferred. See **Chapter 57, Access Control Lists Introduction** for further access list overview and configuration information.

Use the **no** variant of this command in Router Configuration mode to disable this feature for networks as defined in an associated access-list.

Syntax

```
distribute-list <access-list> in

no distribute-list [<access-list>] in

distribute-list <access-list> out {connected|ospf [<process-tag>]|
   rip|static}

no distribute-list <access-list> out {connected|ospf [<process-tag>]|
   rip|static}
```

Parameter	Description
<access-list>	Specifies the IPv6 access-list number or name to use. The specified access list defines which networks are received and which are suppressed.
`in`	Indicates that this applies to incoming advertised routes.
`out`	Indicates that this applies to outgoing advertised routes.
`connected`	Specify the redistribution of connected routes.
`ospf`	Specify the redistribution of OSPFv3 routes.
<process-tag>	Optionally specify an OSPFv3 process tag for OSPFv3 routes.
`rip`	Specify the redistribution of RIPng routes.
`static`	Specify the redistribution of connected routes.

Default Disabled

Mode Router Configuration

Usage This command applies filtering to the transfer of routing information between OSPFv3 and the IPv6 route table. You can apply filtering in either direction, from OSPFv3 to the IPv6 route table using an **in** distribute-list, or from the IPv6 route table to OSPFv3 using an **out** distribute-list.

The effect of an **in** filter is that some route information that OSPFv3 has learned from LSA updates will not be installed into the IPv6 route table. The effect of an **out** filter is that some route information that could be redistributed to OSPFv3 will not be redistributed to OSPFv3.

There are **in** and **out** distribute-lists, which carry out different route filtering activities:

- The **in** distribute list is applied to the process of installing OSPFv3 routes into the IPv6 route table. The SPF calculation generate a set of routes calculated from the LSA database. By default, all of these routes become OSPFv3 candidate routes for inclusion into the IPv6 route table.
- An **in** distribute-list can be used to control whether or not certain routes generated by the SPF calculation are included into the set of candidates for inclusion into the IP route table. Those routes that match **deny** entries in the distribute-list will not be considered for inclusion into the IPv6 route table.
- The **out** distribute-list applies the process of redistributing non-OSPFv3 routes into OSPFv3. If OSPFv3 redistribution is configured, and an **out** distribute-list is also configured, then routes that match deny entries in the distribute-list will not be redistributed into OSPFv3.

Example The below commands redistribute incoming route updates from networks defined with the standard named access-list called `myacl`:

```
awplus# configure terminal
awplus(config)# ipv6 access-list standard myacl permit
2001:db8:1::/64
awplus(config)# router ipv6 ospf
awplus(config-router)# distribute-list myacl in
```

The below commands redistribute outgoing connected route updates from networks defined with the standard named access-list called `myacl`:

```
awplus# configure terminal
awplus(config)# ipv6 access-list standard myacl permit
2001:db8:1::/64
awplus(config)# router ipv6 ospf
awplus(config-router)# distribute-list myacl out connected
```

The below commands disable incoming route updates from networks defined with the standard named access-list called `myacl`:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no distribute-list myacl in
```

The below commands disable outgoing connected route updates from networks defined with the standard named access-list called `myacl`:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no distribute-list myacl out connected
```

Related Commands **ipv6 access-list extended (named)**
ipv6 access-list standard (named)

ipv6 ospf authentication spi

Use this command in Interface Configuration mode to enable either MD5 (Message-Digest 5) or SHA1 (Secure Hash Algorithm 1) authentication for a specified interface.

Use the **no** variant of this command in Interface Configuration mode to disable the authentication configured for a specified interface.

Syntax

```
ipv6 ospf authentication ipsec spi <256-4294967295> {md5 <MD5-key>|
    sha1 <SHA1-key>}

ipv6 ospf authentication null

no ipv6 ospf authentication ipsec spi <256-4294967295>
```

Parameter	Description
`authentication`	Specify this keyword to enable authentication.
`ipsec`	Specify this keyword to use IPsec authentication.
`spi`	Specify this keyword to set the SPI (Security Parameters Index).
`<256-4294967295>`	Specify an SPI (Security Parameters Index) value in the range 256 to 4294967295, entered as a decimal integer.
`md5`	Specify the MD5 (Message-Digest 5) hashing algorithm.
`<MD5-key>`	Enter an MD5 key containing up to 32 hexadecimal characters.
`sha1`	Specify the SHA-1 (Secure Hash Algorithm 1) hashing algorithm.
`<SHA1-key>`	Enter an SHA-1 key containing up to 40 hexadecimal characters.
`null`	Specify no authentication is applied when no other parameters are applied after this keyword (`ipv6 ospf authentication null`). Note this overrides any existing area authentication configured.

Mode Interface Configuration

Default Authentication is not configured on an interface by default.

Usage Configure the same SPI (Security Parameters Index) value on all interfaces that connect to the same link. SPI values are used by link interfaces. Use a different SPI value for a different link interface when using OSPFv3 with link interfaces.

Use the **sha1** keyword to choose SHA-1 authentication instead of entering the **md5** keyword to use MD5 authentication. The SHA-1 algorithm is more secure than the MD5 algorithm. SHA-1 uses a 40 hexadecimal character key instead of a 32 hexadecimal character key as used for MD5 authentication.

Use the **null** keyword to override existing area authentication. Apply the null keyword if area authentication is already configured to configure authentication on an interface.

Use the **null** keyword to override existing area authentication. Apply the **null** keyword if area authentication is already configured to configure authentication on an interface.

See the section **OSPFv3 Authentication and Encryption Overview** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption** in

Chapter 43, OSPFv3 for IPv6 Introduction and Configuration for configuration examples and topologies.

Note

You can configure an authentication security policy (SPI) on a VLAN interface with this command, or an OSPFv3 area with the **area authentication ipsec spi** command.

When you configure authentication for an area, the security policy is applied to all VLAN interfaces in the area. Allied Telesis recommends a different authentication security policy is applied to each interface for higher security.

If you apply the `ipv6 ospf authentication null` command this affects authentication configured on both the VLAN interface and the OSPFv3 area.

This is due to OSPFv3 hello messages ingressing VLAN interfaces, which are part of area authentication, not being authenticated. So neighbors time out.

Example To enable MD5 authentication with a 32 hexadecimal character key for interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# area 1 authentication ipsec spi 1000 md5
1234567890ABCDEF1234567890ABCDEF
```

To enable SHA-1 authentication with a 32 hexadecimal character key for interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf authentication ipsec spi 1000
sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To specify no authentication is applied to interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf authentication null
```

To disable authentication for interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 ospf authentication ipsec spi 1000
```

Related Commands

area authentication ipsec spi
area encryption ipsec spi esp
ipv6 ospf encryption spi esp
show ipv6 ospf interface

ipv6 ospf cost

This command explicitly specifies the cost of the link-state metric in a router-LSA.

The interface cost indicates the overhead required to send packets across a certain VLAN interface. Use this command to set the VLAN interface cost manually.

The **no** variant of this command resets the VLAN interface cost to the default.

Syntax

```
ipv6 ospf cost <1-65535>
no ipv6 ospf cost
```

Parameter	Description
<1-65535>	The link-state metric.

Default By default there is no static value set and the OSPF cost is automatically calculated by using the command **auto-cost reference bandwidth (IPv6 OSPF)** command on page 44.19.

Mode Interface Configuration for a VLAN interface.

Usage This command explicitly sets a user specified cost of sending packets out the interface. Using this command overrides the cost value calculated automatically with the auto-cost reference bandwidth (IPv6 OSPF) feature.

The link-state metric cost is stated in the Router-LSA's link. Typically, the cost is inversely proportional to the bandwidth of an interface. By default, the cost of a VLAN interface is calculated according to the following formula:

reference bandwidth / interface bandwidth

The reference bandwidth is set by default at 1000000 kbps (or 1000 Mbps), but can be changed by the **auto-cost reference bandwidth (IPv6 OSPF)** command on page 44.19.

The interface bandwidth is set by default to 1000000 kbps (or 1000 Mbps), but can be changed by the **bandwidth (duplicate)** command on page 44.21.

Example To set the IPv6 OSPF cost to 10 on the VLAN interface `vlan25`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan25
awplus(config-if)# ipv6 ospf cost 10
```

Related Commands

show ipv6 ospf interface
auto-cost reference bandwidth (IPv6 OSPF)
bandwidth (duplicate)

ipv6 ospf dead-interval

This command sets the interval during which no hello packets are received and after which a neighbor is declared dead.

The dead-interval is the amount of time that OSPF waits to receive an OSPF hello packet from the neighbor before declaring the neighbor is down. This value is advertised in the router's hello packets. It must be a multiple of the hello-interval and be the same for all routers on a specific network.

The **no** variant of this command returns the interval to the default of 40 seconds.

Syntax

```
ipv6 ospf dead-interval <1-65535> [<inst-id>]
no ipv6 ospf  dead-interval
```

Parameter	Description
<1-65535>	The interval in seconds. Default: `40`
<inst-id>	The instance ID Default: `0`

Mode Interface Configuration for a VLAN interface.

Example The following example shows configuring the dead-interval to 10 seconds on the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf dead-interval 10
```

Related Commands

ipv6 ospf hello-interval
show ipv6 ospf interface

ipv6 ospf display route single-line

Use this command to change the result of the **show ipv6 route** command to display each route entry on a single line.

Syntax

```
ipv6 ospf display route single-line
no ipv6 ospf display route single-line
```

Mode Global Configuration

Example To display each route entry on a single line.

```
awplus# configure terminal
awplus(config)# ipv6 ospf display route single-line
```

Related Commands show ipv6 ospf route

ipv6 ospf encryption spi esp

Use this command in Interface Configuration mode to enable either AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) or 3DES (Triple Data Encryption Standard) ESP (Encapsulating Security Payload) encryption for a specified interface.

Use the **no** variant of this command in Interface Configuration mode to disable the encryption configured for a specified interface.

Syntax

```
ipv6 ospf encryption ipsec spi <256-4294967295> esp
   {aes-cbc <AES-CBC-key>|3des <3DES-key>|null}{md5 <MD5-key>|
   sha1 <SHA1-key>}

ipv6 ospf encryption null

no ipv6 ospf encryption ipsec spi <256-4294967295>
```

Parameter	Description
<256-4294967295>	Specify an SPI (Security Parameters Index) value in the range 256 to 4294967295, entered as a decimal integer.
`esp`	Specify the esp keyword (Encapsulating Security Payload) to then apply either AES-CBC or 3DES encryption.
`aes-cbc`	Specify this keyword to enable AES-CBC (Advanced Encryption Standard-Cipher Block Chaining) encryption.
<AES-CBC-key>	Enter an AES-CBC key containing either 32, 48, or 64 hexadecimal characters.
`3des`	Specify 3DES (Triple Data Encryption Standard) encryption.
<3DES-key>	Enter a 3DES key containing 48 hexadecimal characters.
`null`	Specify ESP without AES-CBC or 3DES encryption applied.
`md5`	Specify the MD5 (Message-Digest 5) encryption algorithm.
<MD5-key>	Enter an MD5 key containing 32 hexadecimal characters.
`sha1`	Specify the SHA-1 (Secure Hash Algorithm 1) encryption algorithm.
<SHA1-key>	Enter an SHA-1 key containing 40 hexadecimal characters.
`null`	Specify no encryption is applied when no other parameters are applied after this keyword (`ipv6 ospf encryption null`).

Default Authentication is not configured on an interface by default.

Mode Interface Configuration

Usage When you issue this command, authentication and encryption are both enabled. Configure the same SPI (Security Parameters Index) value on all interfaces that connect to the same link. SPI values are used by link interfaces. Use a different SPI value for a different link interface when using OSPFv3 with link interfaces.

Security is achieved using the IPv6 ESP extension header. The IPv6 ESP extension header is used to provide confidentiality, integrity, authentication, and confidentiality. Authentication fields are removed from OSPF for IPv6 packet headers, so applying IPv6 ESP extension headers are required for integrity, authentication, and confidentiality.

Use the **null** keyword to override existing area encryption. Apply the **null** keyword if area encryption is already configured to then configure encryption on an interface instead.

Use the **sha1** keyword to choose SHA-1 authentication instead of entering the **md5** keyword to use MD5 authentication. The SHA-1 algorithm is more secure than the MD5 algorithm. SHA-1 uses a 40 hexadecimal character key instead of a 32 hexadecimal character key as used for MD5 authentication.

See the section **OSPFv3 Authentication and Encryption Overview** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for configuration examples and topologies.

Note

You can configure an encryption security policy (SPI) on a VLAN interface with this command, or an OSPFv3 area with the **area encryption ipsec spi esp** command.

When you configure encryption for an area, the security policy is applied to all VLAN interfaces in the area. Allied Telesis recommends a different encryption security policy is applied for each interface for higher security.

If you apply the `ipv6 ospf encryption null` command this affects encryption configured on both the VLAN interface and the OSPFv3 area.

This is due to OSPFv3 hello messages ingressing VLAN interfaces, which are part of area encryption, not being being encrypted. So neighbors time out.

Example To enable ESP encryption, but not apply an AES-CBC key or a 3DES key, for interface VLAN 2 and MD5 authentication with a 32 hexadecimal character key, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf encryption ipsec spi 1000 esp
null md5
1234567890ABCDEF1234567890ABCDEF
```

To enable ESP encryption, but not apply an AES-CBC key or a 3DES key, for interface VLAN 2 and SHA-1 authentication with a 40 hexadecimal character key, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf encryption ipsec spi 1000 esp
null sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To enable ESP encryption with an 3DES key with a 48 hexadecimal character key and MD5 authentication with a 32 hexadecimal character key for interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf encryption ipsec spi 1000 esp
3des
1234567890ABCDEF1234567890ABCDEF12345678
90ABCDEF md5
1234567890ABCDEF1234567890ABCDEF
```

To enable ESP encryption with an AES-CBC key with a 32 hexadecimal character key and SHA-1 authentication with a 40 hexadecimal character key for interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf encryption ipsec spi 1000 esp
aes-cbc 1234567890ABCDEF1234567890ABCDEF
sha1
1234567890ABCDEF1234567890ABCDEF12345678
```

To specify no ESP encryption is applied to interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf encryption null
```

To disable ESP encryption for interface VLAN 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 ospf encryption ipsec spi 1000
```

Related Commands

area authentication ipsec spi
area encryption ipsec spi esp
ipv6 ospf authentication spi
show ipv6 ospf interface

ipv6 ospf hello-interval

This command specifies the interval between hello packets.

The hello-interval is advertised in the hello packets. Configure the same hello-interval for all routers on a specific network. A shorter interval ensures faster detection of topological changes, but results in more routing traffic.

The **no** variant of this command returns the interval to the default of 10 seconds.

Syntax

```
ipv6 ospf hello-interval <1-65535>
no ipv6 ospf hello-interval
```

Parameter	Description
<1-65535>	The hello-interval in seconds. Default: `10`

Default The default interval is 10 seconds.

Mode Interface Configuration for a VLAN interface.

Example The following example shows setting the `hello-interval` to 3 seconds on the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf hello-interval 3
```

Related Commands **ipv6 ospf dead-interval**
show ipv6 ospf interface

ipv6 ospf network

This command configures the OSPF network type to a type different from the default for the particular VLAN interface.

The **no** variant of this command returns the network type to the default for the particular VLAN interface.

Syntax

```
ipv6 ospf network [broadcast | non-broadcast | point-to-point |
   point-to-multipoint]

no ipv6 ospf network
```

Parameter	Description
`broadcast`	Sets the network type to broadcast.
`non-broadcast`	Sets the network type to NBMA.
`point-to-multipoint`	Sets the network type to point-to-multipoint.
`point-to-point`	Sets the network type to point-to-point.

Default The default is the `broadcast` OSPF network type for a VLAN interface.

Mode Interface Configuration for a VLAN interface.

Usage This command forces the interface network type to the specified type. Depending on the network type, OSPF changes the behavior of the packet transmission and the link description in LSAs.

Example The following example shows setting the network type to `point-to-point` on the VLAN interface `vlan1`:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ipv6 ospf network point-to-point
```

ipv6 ospf priority

This command sets the router priority, which is a parameter used in the election of the designated router for the link.

The **no** variant of this command returns the router priority to the default of 1.

Syntax

```
ipv6 ospf priority <priority>
no ipv6 ospf priority
```

Parameter	Description
<priority>	*<0-255>* Specifies the router priority of the interface. The larger the value, the greater the priority level. The value 0 defines that the switch cannot become either the DR, or backup DR for the link.

Default The default priority is 1.

Mode Interface Configuration for a VLAN interface.

Usage Set the priority to help determine the OSPF Designated Router (DR) for a link. If two routers attempt to become the DR, the router with the higher router priority becomes the DR. If the router priority is the same for two routers, the router with the higher router ID takes precedence.

Routers with zero router priority values cannot become the designated or backup designated router.

Example The following example shows setting the OSPFv3 priority value to 3 on the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf priority 3
```

ipv6 ospf retransmit-interval

Use this command to specify the time between link-state advertisement (LSA) retransmissions for adjacencies belonging to the interface.

Use the **no** variant of this command to return to the default of 5 seconds.

Syntax

```
ipv6 ospf retransmit-interval <1-65535>
no ipv6 ospf retransmit-interval
```

Parameter	Description
<1-65535>	Specifies the interval in seconds.

Default The default interval is 5 seconds.

Mode Interface Configuration for a VLAN interface.

Usage After sending an LSA to a neighbor, the router keeps the LSA until it receives an acknowledgment. In case the router does not receive an acknowledgment during the set time (the retransmit interval value) it retransmits the LSA. Set the retransmission interval value conservatively to avoid needless retransmission. The interval should be greater than the expected round-trip delay between two routers.

Example The following example shows setting the `ospf retransmit interval` to 6 seconds on the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf retransmit-interval 6
```

ipv6 ospf transmit-delay

Use this command to set the estimated time it takes to transmit a link-state-update packet on the VLAN interface.

Use the **no** variant of this command to return to the default of 1 second.

Syntax

```
ipv6 ospf transmit-delay <1-65535>
no ipv6 ospf transmit-delay
```

Parameter	Description
<1-65535>	Specifies the time, in seconds, to transmit a link-state update.

Default The default interval is 1 second.

Mode Interface Configuration for a VLAN interface.

Usage The transmit delay value adds a specified time to the age field of an update. If the delay is not added, the time in which the LSA transmits over the link is not considered. This command is especially useful for low speed links. Add transmission and propagation delays when setting the transmit delay value.

Example To set the IPv6 OSPF transmit delay time to 3 seconds on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 ospf transmit-delay 3
```

ipv6 router ospf area

Use this command to enable IPv6 OSPF routing on an interface.

Use the **no** variant of this command to disable IPv6 OSPF routing on an interface.

Syntax

```
ipv6 router ospf area <area-id> [tag <process-id>]
   [instance <inst-id>]

no ipv6 router ospf area <area-id>
```

Parameter	Description
<area-id>	The ID of the IPv6 OSPF routing area. Can be entered as either an IPv4 A.B.C.D address format, or as an unsigned integer in the range, 0 to 4294967295. Use either of the following forms when entering an area-ID: ■ `area-id` <*A.B.C.D*> where A.B.C.D is a number entered in IPv4 address format. ■ `area-id` <0 to 4294967295>.
<process-id>	The process tag denotes a separate router process. It can comprise any string of alphanumeric characters. Note that this tag is local to the router on which it is set and does not appear in any OSPF packets or LSA.
<instance-id>	The OSPF instance ID, entered as an integer between 0 and 255. This is the value that will appear in the instance field of the IPv6 OSPF hello packet, see **Figure 43-2 on page 43.6**.

Defaults IPv6 OSPF routing is disabled by default.

When enabling IPv6 OSPF routing:

- the process-tag will default to a null value if not set.
- the Instance ID defaults to 0 if not set.

Mode Interface Configuration for a VLAN interface.

Usage When enabling IPv6 OSPF routing on an interface, specifying the area-ID is mandatory, but the Process tag and Instance are optional.

For more information see **"Configuring instances and processes" on page 43.31**.

Examples The following commands enable IPv6 OSPF on VLAN interface `vlan2`, OSPF area 1, tag PT2, and instance 2.:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 router ospf area 1 tag PT2 instance-id 2
```

The following commands disable IPv6 OSPF on VLAN interface `vlan2` and OSPF area 1:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 router ospf area 1
```

max-concurrent-dd (IPv6 OSPF)

Use this command to limit the number of neighbors that can be concurrently processed in the database exchange. The specified value limits the number of neighbors from all interfaces, not per interface.

Use the **no** variant of this command to have no limit on the maximum number of LSAs.

Syntax

```
max-concurrent-dd <max-neighbors>
no max-concurrent-dd
```

Parameter	Description
<max-neighbors>	<1-65535> The maximum number of neighbors.

Mode Router Configuration

Usage This command is useful where bringing up several adjacencies on a router is affecting performance. In this situation, you can often enhance the system performance by limiting the number of neighbors that can be processed concurrently.

Example The following example sets the max-concurrent-dd value to allow only 4 neighbors to be processed at a time.

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# max-concurrent-dd 4
```

Related Commands **router ipv6 ospf**

passive-interface (IPv6 OSPF)

Use this command to suppress the sending of Hello packets on a specified interface. If you use the **passive-interface** command without the optional parameters then **all** interfaces are put into passive mode.

Use the **no** variant of this command to allow the sending of Hello packets on all interfaces, or on the specified interface. If you use the **no** variant of this command without the optional parameters then **all** interfaces are removed from passive mode.

Syntax

```
passive-interface [<interface>]
no passive-interface [<interface>]
```

Parameter	Description
<interface>	The name or the VID of the VLAN interface.

Mode Router Configuration

Usage Configure an interface to be passive if you wish its connected route to be treated as an OSPF route (rather than an AS-external route), but do not wish to actually exchange any OSPF packets via this interface.

Examples To configure passive interface mode on interface **vlan2**, enter the following commands:

```
awplus(config)# router ipv6 ospf
awplus(config-router)# passive-interface vlan2
```

To configure passive interface mode on **all** interfaces, enter the following commands:

```
awplus(config)# router ipv6 ospf
awplus(config-router)# passive-interface
```

To remove passive interface mode on interface **vlan2**, enter the following commands:

```
awplus(config)# router ipv6 ospf
awplus(config-router)# no passive-interface vlan2
```

To remove passive interface mode on **all** interfaces, enter the following commands:

```
awplus(config)# router ipv6 ospf
awplus(config-router)# no passive-interface
```

redistribute (IPv6 OSPF)

Use this command to redistribute routes from other routing protocols, static routes and connected routes into an IPv6 OSPF routing table.

Use the **no** variant of this command to disable this function.

Syntax

```
redistribute <protocol>} [metric <0-16777214>] [metric-type {1|2}]
   [route-map <route-map-entry>]

no redistribute <protocol>
```

Parameter	Description	
<protocol>	The routing protocol to be redistributed, can be one of:	
	`connected`	Connected routes
	`rip`	Routing Internet Protocol
	`static`	Static Routes
`metric`	*<0-16777214>* Specifies the external metric.	
`metric-type`	Specifies the external metric-type, either type 1 or type 2. ■ **For Metric Type 1:** The best route is based on the external redistributed path cost plus the internal path cost presented by the native routing protocol. ■ **For Metric Type 2:** The best route is based only on the external redistributed path cost. The internal path cost is only used to break a "tie" situation between two identical external path costs.	
`route-map`	*<route-map-entry>* Where the route-map-entry specifies the pointer to the specific route-map.	

Default The default metric value for routes redistributed into OSPFv3 is 20. The metric can also be defined using the **set metric** command for a route map. Note that a metric defined using the **set metric** command for a route map overrides a metric defined with this command.

Mode Router Configuration

Usage IPv6 OSPF advertises routes learned from the RIP routing protocol including static or connected routes. Each injected prefix is put into the AS-external-LSA with a specified metric and metric type.

See the section **OSPFv3 Metrics** in the **OSPFv3 for IPv6 Introduction and Configuration** chapter for more information about metrics, and about behavior when configured in route maps.

Example The following example shows the redistribution of RIP routes into the IPv6 OSPF routing table, with a metric of 10 and a metric type of 1.

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# redistribute rip metric 10 metric-type 1
```

restart ipv6 ospf graceful

Use this command to force the OSPFv3 process to restart. You may optionally specify a grace-period value. If a grace-period is not specified then a default value of 120 seconds is applied.

You should specify a grace-period value of 120 seconds or more. Low grace-period values may cause the graceful restart process on neighboring routers to terminate with routes missing.

Syntax `restart ipv6 ospf graceful [grace-period <1-1800>]`

Parameter	Description
`grace-period`	Specify the grace period.
<1-1800>	The grace period in seconds.

Default The default OSPF grace-period is 120 seconds.

Mode Privileged Exec

Usage After this command is executed, the OSPFv3 process immediately shuts down. It notifies the system that OSPF has performed a graceful shutdown. Routes installed by OSPF are preserved until the grace-period expires.

When a **restart ospf graceful** command is issued, the OSPF configuration is reloaded from the last saved configuration. Ensure you first enter the **copy running-config startup-config** command.

Example To restart OSPFv3, use the following commands:

```
awplus# copy running-config startup-config
awplus# restart ipv6 ospf graceful grace-period 200
```

To apply the default grace-period (120 seconds), use the following commands:

```
awplus# copy running-config startup-config
awplus# restart ipv6 ospf graceful
```

router ipv6 ospf

Use this command to create or remove an IPv6 OSPF routing process, or to enter the Router Configuration mode to configure a specific IPv6 OSPF routing process. Use the **no** variant of this command to terminate an IPv6 OSPF routing process.

Use the **no** parameter with the **process-id** parameter, to terminate and delete a specific IPv6 OSPF routing process.

Syntax

```
router ipv6 ospf [<process-id>]
no router ipv6 ospf [<process-id>]
```

Parameter	Description
<process-id>	A character string that identifies a routing process. If you do not specify the process-id a "null" process ID will be applied. Note that this will appear in show output as *null* However you cannot select the null process by using the character string *null* as command entry characters.

Default No routing process is defined by default.

Mode Global Configuration

Usage The process ID enables you to run more than one OSPF session within the same router, then configure each session to a different router port. Note that this function is internal to the router, and other routers (neighbors) have no knowledge of these different processes. The hello and LSAs issued from each process will appear as if coming from a separate physical router.

To a large extent the requirement for multiple processes has been replaced by the ability within IPv6 OSPF of running simultaneous router instances.

The process ID of IPv6 OSPF is an optional parameter for the **no** variant of this command only. When removing all IPv6 OSPF processes on the switch, you do not need to specify each Process ID, but when removing particular IPv6 OSPF processes, you must specify each Process ID to be removed.

For a description of processes and instances and their configuration relationships, see **"Configuring instances and processes" on page 43.31**.

Example This example shows the use of this command to enter Router Configuration mode.

```
awplus# configure terminal
awplus(config)# router ipv6 ospf P100
awplus(config-router)#
```

router-id (IPv6 OSPF)

Use this command to specify a router ID for the IPv6 OSPF process.

Use the **no** variant of this command to disable this function.

Syntax

```
router-id <router-id>
no router-id
```

Parameter	Description
<router-id>	Specifies the router ID in IPv4 address format.

Mode Router Configuration

Usage Configure each router with a unique router-id. In an IPv6 OSPF router process that has active neighbors, a new router-id takes effect at the next reload or when you restart OSPF manually.

Example The following example shows a specified router ID 0.0.4.5.

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# router-id 0.0.4.5
```

Related Commands show ipv6 ospf

show debugging ipv6 ospf

Use this command in User Exec or Privileged Exec modes to display which OSPFv3 debugging options are currently enabled.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging ipv6 ospf`

Mode User Exec and Privileged Exec

Example

```
awplus# show debugging ipv6 ospf
```

Output **Figure 44-1: Example output from the show debugging ipv6 ospf command**

```
OSPFv3 debugging status:
  OSPFv3 all packet detail debugging is on
  OSPFv3 all IFSM debugging is on
  OSPFv3 all NFSM debugging is on
  OSPFv3 all LSA debugging is on
  OSPFv3 all NSM debugging is on
  OSPFv3 all route calculation debugging is on
  OSPFv3 all event debugging is on
```

show ipv6 ospf

Use this command in User Exec or Privileged Exec modes to display general information about all IPv6 OSPF routing processes, including OSPFv3 Authentication configuration and status information.

Include the process ID parameter with this command to display information about specified processes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf
show ipv6 ospf <process-id>
```

Parameter	Description
<process-id>	*<0-65535>* The ID of the router process for which information will be displayed. If this parameter is included, only the information for the specified routing process is displayed.

Mode User Exec and Privileged Exec

Examples To display general information about all IPv6 OSPF routing processes, use the command:

```
awplus# show ipv6 ospf
```

To display general information about IPv6 OSPF (OSPFv3) routing process `P10`, use the command:

```
awplus# show ipv6 ospf P10
```

Output **Figure 44-2: Example output from the show ipv6 ospf command for process P10 showing OSPFv3 Authentication configuration information highlighted in bold**

```
 awplus#show ipv6 ospf
  Routing Process "OSPFv3 (10)" with ID 192.168.1.2
  Route Licence: Route : Limit=Unlimited, Allocated=0, Visible=0,
Internal=0
  Route Licence: Breach: Current=0, Watermark=0
  Process uptime is 6 minutes
  Current grace period is 120 secs (default)
  SPF schedule delay min 0.500 secs, SPF schedule delay max 50.0
secs
  Minimum LSA interval 5 secs, Minimum LSA arrival 1 secs
  Number of incoming current DD exchange neighbors 0/5
  Number of outgoing current DD exchange neighbors 0/5
  Number of external LSA 0. Checksum Sum 0x0000
  Number of AS-Scoped Unknown LSA 0
  Number of LSA originated 4
  Number of LSA received 10
  Number of areas in this router is 1
     Area BACKBONE(0)
         Number of interfaces in this area is 1(1)
         MD5 Authentication SPI 1000
         NULL Encryption SHA-1 Auth, SPI 1001
         SPF algorithm executed 9 times
         Number of LSA 3.  Checksum Sum 0xF9CC
         Number of Unknown LSA 0
```

Related Commands

area authentication ipsec spi
area encryption ipsec spi esp
router ipv6 ospf

show ipv6 ospf database

Use this command in User Exec or Privileged Exec modes to display a database summary for IPv6 OSPF information. Include the process ID parameter with this command to display information about specified processes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 ospf <process-id> database [self-originate|max-age|adv router <adv-router-id>]`

Parameter	Description
<process-id>	*<0-65535>* The ID of the router process for which information will be displayed.
`self-originate`	Displays self-originated link states.
`max-age`	Displays LSAs in MaxAge list. It maintains the list of the all LSAs in the database which have reached the max-age which is 3600 seconds.
`adv-router`	Advertising Router LSA.
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.

Mode User Exec and Privileged Exec

Example To display the database summary for IPv6 OSPF information on process `P10`, use the command:

```
awplus# show ipv6 ospf P10 database
```

Output

Figure 44-3: Example output from the show ipv6 ospf P10 database command

```
OSPFv3 Router with ID (0.0.1.1) (Process P10)

                Link-LSA (Interface vlan2)

Link State ID   ADV Router      Age  Seq#       CkSum  Prefix
0.0.0.202       0.0.1.1          46 0x800000c3 0x5f50      1
0.0.0.202       0.0.1.2           8 0x800000c3 0x4ca0      1

                Link-LSA (Interface vlan3)

Link State ID   ADV Router      Age  Seq#       CkSum  Prefix
0.0.0.203       0.0.1.1         1071 0x8000000e 0xe082      1
0.0.0.203       0.0.1.3         1057 0x8000000e 0xb8aa      1

                Router-LSA (Area 0.0.0.0)

Link State ID   ADV Router      Age  Seq#       CkSum    Link
0.0.0.0         0.0.1.1         1016 0x800000cd 0xa426      2
0.0.0.0         0.0.1.2          979 0x800000d8 0xad2b      1
0.0.0.0         0.0.1.3         1005 0x800000cf 0xefed      1

                Network-LSA (Area 0.0.0.0)

Link State ID   ADV Router      Age  Seq#       CkSum
0.0.0.202       0.0.1.2         1764 0x800000c2 0x94c3
0.0.0.203       0.0.1.3         1010 0x800000c4 0x8ac8

                Intra-Area-Prefix-LSA (Area 0.0.0.0)

Link State ID   ADV Router      Age  Seq#       CkSum  Prefix  Reference
0.0.0.2         0.0.1.2          978 0x800000a1 0x699a      1  Router-LSA
0.0.0.4         0.0.1.2         1764 0x800000c2 0xca4d      1  Network-LSA
0.0.0.1         0.0.1.3         1004 0x80000012 0xaee2      1  Router-LSA
0.0.0.7         0.0.1.3         1005 0x8000000e 0x3c89      1  Network-LSA

                AS-external-LSA

Link State ID   ADV Router      Age  Seq#       CkSum
0.0.0.13        0.0.1.1         1071 0x8000000e 0xca9f E2
0.0.0.14        0.0.1.1         1071 0x8000000e 0xcc9b E2
0.0.0.15        0.0.1.1         1071 0x8000000e 0xce97 E2
0.0.0.16        0.0.1.1         1071 0x8000000e 0xd093 E2
0.0.0.17        0.0.1.1         1071 0x8000000e 0xd28f E2
0.0.0.18        0.0.1.1         1071 0x8000000e 0xd48b E2
```

show ipv6 ospf database external

Use this command in User Exec or Privileged Exec modes to display information about the external LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database external <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.
`self originate`	Self-originated link states.
`adv-router`	Displays all the LSAs of the specified router.

Mode User Exec and Privileged Exec

Examples To display information about the external LSAs, use the following command:

```
awplus# show ipv6 ospf database external adv-router 10.10.10.1
```

Output **Figure 44-4: Example output from the show ipv6 ospf database external command**

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf database grace

Use this command in User Exec or Privileged Exec modes to display information about the grace LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database grace <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.
`adv-router`	Displays all the LSAs of the specified router.
`self originate`	Self-originated link states.

Mode User Exec and Privileged Exec

Examples To display information about the grace LSAs, use the following command:

```
awplus# show ipv6 ospf database grace adv-router 10.10.10.1
```

Output **Figure 44-5: Example output from the show ipv6 ospf database grace command**

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf database inter-prefix

Use this command in User Exec or Privileged Exec modes to display information about the inter-prefix LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database inter-prefix <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.
`adv-router`	Displays all the LSAs of the specified router.
`self originate`	Self-originated link states.

Mode User Exec and Privileged Exec

Examples To display information about the inter-prefix LSAs, use the following command:

```
awplus# show ipv6 ospf database external adv-router 10.10.10.1
```

Output **Figure 44-6: Example output from the show ipv6 ospf database inter-prefix command**

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf database inter-router

Use this command in User Exec or Privileged Exec modes to display information about the inter-router LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database inter-router <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.
`adv-router`	Displays all the LSAs of the specified router.
`self originate`	Self-originated link states.

Mode User Exec and Privileged Exec

Examples To display information about the inter-router LSAs, use the following command:

```
awplus# show ipv6 ospf database inter-router adv-router
       10.10.10.1
```

Output **Figure 44-7: Example output from the show ipv6 ospf database inter-router command**

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf database intra-prefix

Use this command in User Exec or Privileged Exec modes to display information about the intra-prefix LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database intra-prefix <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.
`adv-router`	Displays all the LSAs of the specified router.
`self originate`	Self-originated link states.

Mode User Exec and Privileged Exec

Examples To display information about the intra-prefix LSAs, use the following command:

```
awplus# show ipv6 ospf database intra-prefix adv-router
       10.10.10.1
```

Output **Figure 44-8: Example output from the show ipv6 ospf database intra-prefix command**

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf database link

Use this command in User Exec or Privileged Exec modes to display information about the link LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database link <adv-router-id> [self-originate|adv-
   router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The Advertising Router ID (usually entered in IPv4 address format A.B.C.D). Note that this ID component no longer represents an address; it is simply a character string that has an IPv4 address format.
`adv-router`	Displays all the LSAs of the specified router.
`self originate`	Self-originated link states.

Mode User Exec and Privileged Exec

Examples To display information about the link LSAs, use the following command:

```
awplus# show ipv6 ospf database link adv-router 10.10.10.1
```

Output **Figure 44-9: Example output from the show ipv6 ospf database link command**

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


  LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf database network

Use this command in User Exec or Privileged Exec modes to display information about the network LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database network <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The router ID of the advertising router, in IPv4 address format. Note, however, that this no longer represents a real address.
`self-originate`	Self-originated link states.
`adv-router`	The advertising router selected.

Mode User Exec and Privileged Exec

Examples To display information about the OSPFv3 network LSAs, use the following command:

```
awplus# show ipv6 ospf database network
```

Output **Figure 44-10: Example output from the show ipv6 ospf database network command**

```
OSPFv3 Router with ID (0.0.1.1) (Process P10)

                Network-LSA (Area 0.0.0.0)

  LS age: 97
  LS Type: Network-LSA
  Link State ID: 0.0.0.202
  Advertising Router: 0.0.1.2
  LS Seq Number: 0x800000C3
  Checksum: 0x92C4
  Length: 32
  Options: 0x000013 (-|R|-|-|E|V6)
    Attached Router: 0.0.1.2
    Attached Router: 0.0.1.1

  LS age: 1144
  LS Type: Network-LSA
  Link State ID: 0.0.0.203
  Advertising Router: 0.0.1.3
  LS Seq Number: 0x800000C4
  Checksum: 0x8AC8
  Length: 32
  Options: 0x000013 (-|R|-|-|E|V6)
    Attached Router: 0.0.1.3
    Attached Router: 0.0.1.1
```

show ipv6 ospf database router

Use this command in User Exec or Privileged Exec modes to display information only about the router LSAs.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf database router <adv-router-id> [self-originate|
   adv-router <adv-router-id>]
```

Parameter	Description
<adv-router-id>	The router ID of the advertising router, in IPv4 address format. Note, however, that this no longer represents a real address.
`self-originate`	Self-originated link states.
`adv-router`	The advertising router selected.

Mode User Exec and Privileged Exec

Examples To display information about the OSPFv3 router LSAs, use the following command:

```
awplus# show ipv6 ospf database router
```

Output **Figure 44-11: Example output from the show ipv6 ospf database router command**

```
            OSPFv3 Router with ID (0.0.1.3) (Process P10)

                 Router-LSA (Area 0.0.0.0)

LS age: 556
  LS Type: Router-LSA
  Link State ID: 0.0.0.0
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x800000CA
  Checksum: 0xAA23
  Length: 56
  Flags: 0x02 (-|-|E|-)
  Options: 0x000013 (-|R|-|-|E|V6)

    Link connected to: a Transit Network
      Metric: 1
      Interface ID: 203
      Neighbor Interface ID: 203
      Neighbor Router ID: 0.0.1.3

    Link connected to: a Transit Network
      Metric: 1
      Interface ID: 202
      Neighbor Interface ID: 202
      Neighbor Router ID: 0.0.1.2


  LS age: 520
  LS Type: Router-LSA
  Link State ID: 0.0.0.0
  Advertising Router: 0.0.1.2
  LS Seq Number: 0x800000D5
  Checksum: 0xB328
  Length: 40
  Flags: 0x00 (-|-|-|-)
  Options: 0x000013 (-|R|-|-|E|V6)

    Link connected to: a Transit Network
      Metric: 1
      Interface ID: 202
      Neighbor Interface ID: 202
      Neighbor Router ID: 0.0.1.2


LS age: 543
  LS Type: Router-LSA
  Link State ID: 0.0.0.0
  Advertising Router: 0.0.1.3
  LS Seq Number: 0x800000CC
  Checksum: 0xF5EA
  Length: 40
  Flags: 0x00 (-|-|-|-)
  Options: 0x000013 (-|R|-|-|E|V6)

    Link connected to: a Transit Network
      Metric: 1
      Interface ID: 203
      Neighbor Interface ID: 203
      Neighbor Router ID: 0.0.1.3
            OSPFv3 Router with ID (0.0.1.3) (Process P10)

                 AS-external-LSA
```

```
  LS age: 1384
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x80000009
  Checksum: 0xD49A
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1384
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x80000009
  Checksum: 0xD696
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1384
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x80000009
  Checksum: 0xD892
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.13
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xCE9D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2010:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.14
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD099
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2011:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

```
LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.15
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD295
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2012:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.16
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD491
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2013:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.17
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD68D
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2014:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1


LS age: 1087
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.18
  Advertising Router: 0.0.1.1
  LS Seq Number: 0x8000000C
  Checksum: 0xD889
  Length: 52
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2015:2222::/64
    Prefix Options: 0 (-|-|-|-)
    Forwarding Address: 2003:1111::1
```

show ipv6 ospf interface

Use this command in User Exec or Privileged Exec modes to display interface information for OSPF for all interfaces or a specified interface, including OSPFv3 Authentication status for all interfaces or for a specified interface.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf interface [<interface-name>]
```

Parameter	Description
<interface-name>	An alphanumeric string that is the interface name. Omit the optional interface to display OSPF

Mode User Exec and Privileged Exec

Examples

```
awplus# show ipv6 ospf interface vlan2
```

Output **Figure 44-12: Example output from the show ipv6 ospf interface command showing OSPFv3 Authentication configuration information highlighted in bold**

```
awplus#show ipv6 ospf interface
 vlan2 is up, line protocol is up
   Interface ID 302
   IPv6 Prefixes
     fe80::215:77ff:fead:f87e/64 (Link-Local Address)
   Security Policy
     MD5 Authentication SPI 1000
     NULL Encryption SHA-1 Auth, SPI 1001

   OSPFv3 Process (10), Area 0.0.0.0, Instance ID 0
     Router ID 192.168.1.2, Network Type BROADCAST, Cost: 1
     Transmit Delay is 1 sec, State Backup, Priority 1
     Interface state Backup
     Designated Router (ID) 192.168.1.1
       Interface Address fe80::21d:e5ff:fec9:cfbe
     Backup Designated Router (ID) 192.168.1.2
       Interface Address fe80::215:77ff:fead:f87e
     Timer interval configured, Hello 10, Dead 40, Wait 40,
Retransmit 5
       Hello due in 00:00:07
     Neighbor Count is 1, Adjacent neighbor count is 1
```

Figure 44-13: Example output from the show ipv6 ospf interface vlan3 command

```
awplus#show ipv6 ospf interface vlan3
vlan3 is up, line protocol is up
  Interface ID 203
  IPv6 Prefixes
    fe80::200:cdff:fe24:daae/64 (Link-Local Address)
    2003:1111::2/64
  OSPFv3 Process (P1), Area 0.0.0.0, Instance ID 0
    Router ID 0.0.1.1, Network Type BROADCAST, Cost: 1
    Transmit Delay is 1 sec, State DR, Priority 1
    Designated Router (ID) 0.0.1.1
      Interface Address fe80::200:cdff:fe24:daae
    No backup designated router on this link
    Timer interval configured, Hello 10, Dead 40, Wait 40,
Retransmit 5
      Hello due in 00:00:02
    Neighbor Count is 0, Adjacent neighbor count is 0
```

Related Commands ipv6 ospf authentication spi
ipv6 ospf encryption spi esp

show ipv6 ospf neighbor

Use this command in User Exec or Privileged Exec modes to display information on OSPF neighbors. Include the process ID parameter with this command to display information about specified processes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 ospf [<process-id>] neighbor <neighbor-id>
show ipv6 ospf [<process-id>] neighbor detail
show ipv6 ospf [<process-id>] neighbor <interface> [detail]
```

Parameter	Description
<process-id>	*<character string>* The ID of the OSPF process for which information will be displayed.
<neighbor-id>	The Neighbor ID, entered in IP address (A.B.C.D) format.
detail	Detail of all neighbors.
<interface>	IP address of the interface.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ipv6 ospf neighbor
```

Output **Figure 44-14: Example output from the show ipv6 ospf neighbor command**

```
awplus#show ipv6 ospf P1 neighbor 2.2.2.2
OSPFv3 Process (P1)
Neighbor ID     Pri       State                 Dead Time   Interface Instance ID
2.2.2.2         5         2-Way/DROther         00:00:33    vlan3              0
```

Figure 44-15: Example output from the show ipv6 ospf neighbor detail command

```
awplus#show ipv6 ospf neighbor detail
Neighbor 0.0.1.2, interface address fe80::215:77ff:fec9:7472
   In the area 0.0.0.0 via interface vlan2
   Neighbor priority is 1, State is Full, 6 state changes
   DR is 0.0.1.2     BDR is 0.0.1.1
   Options is 0x000013 (-|R|-|-|E|V6)
   Dead timer due in 00:00:33
   Database Summary List 0
   Link State Request List 0
   Link State Retransmission List 0
```

show ipv6 ospf route

Use this command in User Exec or Privileged Exec modes to display the OSPF routing table. Include the process ID parameter with this command to display the OSPF routing table for specified processes.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 ospf [<process-id>] route`

Parameter	Description
<process-id>	A character string that specifies the router process. If this parameter is included, only the information for this specified routing process is displayed.

Mode User Exec and Privileged Exec

Examples To display the OSPF routing table, use the command:

```
awplus# show ipv6 ospf route
```

Output **Figure 44-16: Example output from the show ipv6 ospf P10 route command for a specific process**

```
OSPFv3 Process (P1)
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter
area
       E1 - OSPF external type 1, E2 - OSPF external type 2

   Destination                                      Metric
     Next-hop
O  2002:1111::/64                                      2
     via fe80::200:cdff:fe24:daae, vlan3, Area 0.0.0.0
C  2003:1111::/64                                      1
     directly connected, vlan3, Area 0.0.0.0
O  2004:1111::/64                                      3
     via fe80::200:cdff:fe24:daae, vlan3, Area 0.0.0.0
C  2005:1111::/64                                      1
     directly connected, vlan5, Area 0.0.0.0
E2 2010:2222::/64                                   1/20
     via 2003:1111::1, vlan3
E2 2011:2222::/64                                   1/20
     via 2003:1111::1, vlan3
E2 2012:2222::/64                                   1/20
     via 2003:1111::1, vlan3
E2 2013:2222::/64                                   1/20
     via 2003:1111::1, vlan3
E2 2014:2222::/64                                   1/20
     via 2003:1111::1, vlan3
E2 2015:2222::/64                                   1/20
     via 2003:1111::1, vlan3
```

show ipv6 ospf virtual-links

Use this command in User Exec or Privileged Exec modes to display virtual link information, including OSPFv3 Authentication status for virtual links.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 ospf virtual-links`

Mode User Exec and Privileged Exec

Usage See the section **OSPFv3 Virtual Links** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for introductory OSPFv3 Authentication and Encryption information.

Also see the section **Configuring OSPFv3 Authentication and Encryption for a Virtual Link** in **Chapter 43, OSPFv3 for IPv6 Introduction and Configuration** for configuration examples and topologies.

Examples To display virtual link information, use the command:

```
awplus# show ipv6 ospf virtual-links
```

Output **Figure 44-17: Example output from the show ipv6 ospf virtual-links command showing OSPFv3 Authentication configuration information highlighted in bold**

```
awplus#show ipv6 ospf virtual-links
 Virtual Link VLINK1 to router 192.168.1.10 is down
   Transit area 0.0.0.1 via interface *, instance ID 0
   Local address
   Remote address
   MD5 Authentication SPI 1000
   NULL encryption SHA-1 auth SPI 1001
   Transmit Delay is 1 sec, State Down,
   Timer intervals configured, Hello 10, Dead 40, Wait 40,
Retransmit 5
     Hello due in inactive
     Adjacency state Down
```

Related Commands **area virtual-link authentication ipsec spi**
area virtual-link encryption ipsec spi

summary-address (IPv6 OSPF)

Use this command in Router Configuration mode to summarize, or possibly suppress, external redistributed OSPFv3 routes within the specified address range.

Use the **no** variant of this command in Router Configuration mode to stop summarizing, or suppressing, external redistributed OSPFv3 routes within the specified address range.

Syntax

```
summary-address <ipv6-addr/prefix-length>
   [not-advertise] [tag <0-4294967295>]

no summary-address <ipv6-addr/prefix-length>
   [not-advertise] [tag <0-4294967295>]
```

Parameter	Description
<ipv6-addr/prefix-length>	Specifies the base IPv6 address of the IPv6 summary address. The range of addresses given as IPv6 starting address and an IPv6 prefix length.
`not-advertise`	Set the **not-advertise** option if you do not want OSPFv3 to advertise either the summary address or the individual networks within the range of the summary address.
`tag <0-4294967295>`	The tag parameter specifies the tag value that OSPFv3 places in the AS external LSAs created as a result of redistributing the summary route. The tag overrides tags set by the original route.

Default The default tag value for a summary address is 0.

Mode Router Configuration

Usage An address range is a pairing of an address and a prefix length. Redistributing routes from other protocols into OSPFv3 requires the router to advertise each route individually in an external LSA. Use this command to advertise one summary route for all redistributed routes covered by a specified prefix to decrease the size of the OSPFv3 link state database.

For example, if the specified address range is 2001:0db8:44::/48, then summary-address functionality will match 2001:0db8:4400:0000::1/128 through 2001:0db8:44ff:ffff::1/128.

Ensure OSPFv3 routes exist in the summary address range for advertisement before using this command.

Example The following example uses the `summary-address` command to aggregate external LSAs that match the IPv6 prefix `2001:0db8::/32` and assigns a tag value of `3`.

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# summary-address 2001:0db8::/32 tag 3
```

The following example uses the `no summary-address` command to stop summarizing IPv6 addresses in the address range covered within the IPv6 prefix `2001:0db8::/32`.

```
awplus# configure terminal
awplus(config)# router ipv6 ospf
awplus(config-router)# no summary-address 2001:0db8::/32
```

timers spf (IPv6 OSPF) (deprecated)

Use this command to adjust route calculation timers.

Use the **no** variant of this command to return to the default timer values.

Syntax

```
timers spf <spf-delay> <spf-holdtime>
no timers spf
```

Parameter	Description
<spf-delay>	*<0-2147483647>* Specifies the delay between receiving changed routing information and embarking on an SPF calculation.
<spf-holdtime>	*<0-2147483647>* Specifies the hold time between consecutive SPF calculations.

Default The default spf-delay value is 5 seconds. The default spf-holdtime value is 10 seconds.

Mode Router Configuration

Usage This command configures the delay time between the receipt of a topology change and the calculation of the Shortest Path First (SPF). This command also configures the hold time between two consecutive SPF calculations.

Examples

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# timers spf 7 12
```

Related Commands timers spf exp (IPv6 OSPF)

timers spf exp (IPv6 OSPF)

Use this command to adjust route calculation timers using exponential back-off delays.

Use **no** form of this command to return to the default exponential back-off timer values.

Syntax

```
timers spf exp <min-holdtime> <max-holdtime>
no timers spf exp <min-holdtime> <max-holdtime>
```

Parameter	Description
<min-holdtime>	*<0-2147483647>* Specifies the minimum delay between receiving a change to the SPF calculation in milliseconds. The default SPF min-holdtime value is 50 milliseconds.
<max-holdtime>	*<0-2147483647>* Specifies the maximum delay between receiving a change to the SPF calculation in milliseconds. The default SPF max-holdtime value is 50 seconds.

Mode Router Configuration

Usage This command configures the minimum and maximum delay time between the receipt of a topology change and the calculation of the Shortest Path First (SPF).

Examples To set the minimum delay time to 5 milliseconds and maximum delay time to 10 milliseconds, use the commands:

```
awplus# configure terminal
awplus(config)# router ipv6 ospf 100
awplus(config-router)# timers spf exp 5 20
```

Related Commands **timers spf (IPv6 OSPF) (deprecated)**

undebug ipv6 ospf events

This command applies the functionality of the **no debug ipv6 ospf events** command on page 44.22.

undebug ipv6 ospf ifsm

This command applies the functionality of the **no debug ipv6 ospf ifsm** command on page 44.23.

undebug ipv6 ospf lsa

This command applies the functionality of the **no debug ipv6 ospf lsa** command on page 44.24.

undebug ipv6 ospf nfsm

This command applies the functionality of the **no debug ipv6 ospf nfsm** command on page 44.25.

undebug ipv6 ospf packet

This command applies the functionality of the **no debug ipv6 ospf packet** command on page 44.26.

undebug ipv6 ospf route

This command applies the functionality of the **no debug ipv6 ospf route** command on page 44.27.

Chapter 45: Route Map Configuration

Introduction 45.2
Structure of a Route Map 45.2
Clauses 45.2
Configuring Route Maps for Filtering and Modifying OSPF Routes 45.3
Configuring a Match Clause 45.3
Configuring a Set Clause 45.4
Applying Route Maps in OSPF 45.5

Introduction

This chapter introduces route maps for filtering in OSPF. For details about the commands used in these examples, or the outputs from validation commands, see **Chapter 46, Route Map Commands**.

Route maps offer a complex combination of match criteria and actions. They can be used to filter OSPF routes.

Structure of a Route Map

There are various levels of structure within a route map:

- A route map is an entity with a name
- Each route map consists of multiple entries, identified by sequence numbers
- Each entry can consist of multiple clauses

In effect, an entry defines an individual filter. It can have a **match** clause that defines what it will match on, and it can have multiple **set** clauses that can specify actions to be taken.

A route is matched against each entry in turn. Once an entry is found that matches the route, the action(s) associated with that entry is (are) performed, and no further entries are considered.

For example, if you create an entry that will permit a route, followed by an entry that would deny that route, the route is permitted. As another example, if you create two conflicting set clauses, the first change is applied, not the second.

There is an implicit "match all" filter at the end of the route map. The action on that implicit entry is **deny**. By default, any route that does not explicitly match any particular entry in the route map will be dropped.

You can change this by ending the route map with a "permit all" clause, such as the following:

```
awplus(config)#  route-map <map-name> permit 65535
```

Clauses

There are two types of clauses that can be present in a route map entry:

- match clauses, which specify attributes or prefixes to match on
- set clauses, which specify the changes to be made to attribute values

A given route map entry can never have more than one match clause, but it can have multiple set clauses.

Configuring Route Maps for Filtering and Modifying OSPF Routes

OSPF route maps can be applied to importing:

- OSPF-learned routes into the main IP route table, or
- static or RIP routes into OSPF

Filtering **cannot**:

- remove an entry from the LSA database once the entry has been added
- prevent the switch from advertising an entry to interfaces in the same area that the entry is relevant to
- prevent updates that OSPF learns from being put into the LSA database
- change the properties of an entry in the LSA database

This is because OSPF shares LSAs between all the routers in an area. The protocol assumes that all the routers in the area have shared all the advertisements among each other, and that all agree on the state of the complete link state database for the area. If some routers in the area are learning, but not advertising, that breaks the OSPF model.

Configuring a Match Clause

A match clause can match on the following clauses.

When configuring a match clause, make sure you are in route map mode. The prompt should look like:

```
awplus(config-route-map)#
```

Metric

The entry will match all routes whose metric is equal to that specified in the clause.

To match a metric value, use the command:

```
match metric <value>
```

Interface

The entry will match all routes learned via the specified VLAN.

To match a VLAN, use the command:

```
match interface <vlan>
```

External route type

The entry will match all routes of either Type 1 External or Type 2 External.

To match a route type, use the command:

```
match external {type-1|type-2}
```

A prefix, by using a prefix list

The entry will match one or more route prefixes.

Once you have made the prefix list, apply it to the match clause of a route map entry by using the command:

```
match ip address prefix-list <list-name>
```

A prefix, by using an ACL

An ACL is an alternative to a prefix list for matching route prefixes.

Once you have made the ACL, apply it to the match clause of a route map entry by using the command:

```
match ip address <acl-number-or-name>
```

A next hop address

The entry will match the route's next hop.

You can use either a prefix list or an ACL to specify a next hop address. Once you have made the prefix list or ACL, apply it to the match clause of a route map entry by using one of the commands:

```
match ip next-hop prefix-list <list-name>
```

```
match ip next-hop <acl-number-or-name>
```

Configuring a Set Clause

If a route matches the **match** clause, then the action of the route map entry will be applied to that route. The action might simply be to permit or deny the route, or it might be to update its parameters by applying one or more **set** clauses.

When configuring a set clause, make sure you are in route map mode for the same route map name sequence number as you used for the match clause. The prompt should look like:

```
awplus(config-route-map)#
```

A set clause can alter the following parameters on a route.

Metric This changes the route metric. You can:

- Set the metric, by using the command:

```
set metric <0-4294967295>
```

- Increase or decrease the metric by a specified amount, by using one of the commands:

```
set metric +<amount>
```

```
set metric -<amount>
```

For example, to increase the metric by 2, use the command:

```
set metric +2
```

Note defining the OSPF metric in a route map supersedes the metric defined using a **redistribute (OSPF)** or a **redistribute (IPv6 OSPF)** command.

See the section **OSPF Metrics** in the **OSPF Introduction and Configuration** chapter for more information about OSPF metrics.

See the section **OSPFv3 Metrics** in the **OSPFv3 for IPv6 Introduction and Configuration** chapter for more information about OSPFv3 metrics.

Next hop This specifies the next hop for matching routes.

Use the command:

```
set ip next-hop <ipadd>
```

Type This sets the route type to either Type 1 External or Type 2 External.

Use the command:

```
set metric-type {type-1|type-2}
```

Applying Route Maps in OSPF

To specify a route map to be applied to static, RIP or connected routes as they are imported to OSPF, use the commands:

```
router ospf
redistribute {rip|connected|static} route-map <map-name>
```

Note that if you want to filter OSPF routes as they are imported into the main IP route table, you need to use a distribute filter instead of a route map.

Use commands like the following:

```
router ospf 88
distribute-list list1 in
```

Chapter 46: Route Map Commands

Command List 46.2
match interface 46.3
match ip address 46.4
match ip next-hop 46.6
match ipv6 address 46.8
match metric 46.9
match route-type 46.10
match tag 46.11
route-map 46.12
set ip next-hop (route map) 46.14
set metric 46.15
set metric-type 46.17
set tag 46.18
show route-map 46.19

Command List

This chapter provides an alphabetical reference for route map commands. These commands can be divided into the following categories:

- **route-map** command, used to create a route map and/or route map entry, and to put you into route map mode
- **match** commands, used to determine which routes the route map applies to
- **set** commands, used to modify matching routes

match interface

Use this command to add an interface match clause to a route map entry. Specify the interface name to match.

A route matches the route map if its interface matches the interface name.

Each entry of a route map can only match against one interface in one interface match clause. If the route map entry already has an interface match clause, entering this command replaces that match clause with the new clause.

Use the **no** variant of this command to remove the interface match clause from the route map entry. Use the **no** variant of this command without a specified interface to remove all interfaces.

Syntax

```
match interface <interface>
no match interface [<interface>]
```

Parameter	Description
<interface>	The VLAN to match, e.g. `vlan2`.

Mode Route-map Configuration

Usage This command is valid for RIP and OSPF routes only.

Example To add entry `10` to the route map called `mymap1`, which will process routes if they use the interface `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map mymap1 permit 10
awplus(config-route-map)# match interface vlan1
```

To remove all interfaces from the route map called `mymap1`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map mymap1 permit 10
awplus(config-route-map)# no match interface
```

Related Commands

match ip address
match ip next-hop
match route-type
match tag
route-map
show route-map

match ip address

Use this command to add an IP address prefix match clause to a route map entry. You can specify the prefix or prefixes to match by either:

- specifying the name of an access list. To create the access list, enter Global Configuration mode and use the **access-list** command.
- specifying the name of a prefix list. To create the prefix list, enter Global Configuration mode and use the **ip prefix-list** command.

A route matches the route map entry if the route's prefix matches the access list or prefix list.

Each entry of a route map can have at most one access list-based IP address match clause and one prefix list-based IP address match clause. If the route map entry already has one of these match clauses, entering this command replaces that match clause with the new clause.

Note that access lists, prefix lists and route map entries all specify an action of deny or permit. The action in the access list or prefix list determines whether the route map checks routes for a given prefix. The route map action and its **set** clauses determine what the route map does with routes that contain that prefix.

Use the **no** variant of this command to remove the IP address match clause from a route map entry. To remove a prefix list-based match clause you must also specify the **prefix-list** parameter.

Syntax

```
match ip address {<accesslistID>|prefix-list <prefix-listname>}
no match ip address [<accesslistID>]
no match ip address prefix-list <prefix-listname>
```

Parameter	Description
<accesslistID>	{*<access-list-name>*|<1-199>|<1300-2699>} The IP access list name or number.
<access-list-name>	The IP access list name.
<1-199>	The IP access list number.
<1300-2699>	The IP access list number (expanded range).
prefix-list	Use an IP prefix list to specify which prefixes to match.
<prefix-listname>	The prefix list name.

Mode Route-map Configuration

Usage The `match ip address` command specifies the IP address to be matched. If there is a match for the specified IP address, and `permit` is specified, the route is redistributed or controlled, as specified by the set action. If the match criteria are met, and `deny` is specified then the route is `not` redistributed or controlled. If the match criteria are `not` met, the route is neither accepted nor forwarded, irrespective of `permit` or `deny` specifications.

This command is valid for OSPF and RIP routes.

Examples

To add entry `3` to the route map called `myroute`, which will process routes that match the ACL called `List1`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map myroute permit 3
awplus(config-route-map)# match ip address List1
```

To add entry `3` to the route map called `rmap1`, which will process routes that match the prefix list called `mylist`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# match ip address prefix-list mylist
```

Related Commands

access-list (extended numbered)
access-list (standard numbered)
ip prefix-list
route-map
show ip access-list
show route-map

match ip next-hop

Use this command to add a next-hop match clause to a route map entry. You can specify the next hop to match by either:

- specifying the name of an access list. To create the access list, enter Global Configuration mode and use the **access-list** command.
- specifying the name of a prefix list. To create the prefix list, enter Global Configuration mode and use the **ip prefix-list** command.

A route matches the route map if the route's next hop matches the access list or prefix list.

Each entry of a route map can have at most one access list-based next-hop match clause and one prefix list-based next-hop match clause. If the route map entry already has one of these match clauses, entering this command replaces that match clause with the new clause.

Note that access lists, prefix lists and route map entries all specify an action of deny or permit. The action in the access list or prefix list determines whether the route map checks routes for a given next-hop value. The route map action and its **set** clauses determine what the route map does with routes that contain that next hop.

Use the **no** variant of this command to remove the next-hop match clause from a route map entry. To remove a prefix list-based match clause you must also specify the prefix-list parameter.

Syntax

```
match ip next-hop {<accesslistID>|prefix-list <prefix-listname>}
no match ip next-hop [<accesslistID>]
no match ip next-hop prefix-list [<prefix-listname>]
```

Parameter	Description
<accesslistID>	{*<access-list-name>*|<1-199>|<1300-2699>} The IP access list name or number.
<access-list-name>	The IP access list name.
<1-199>	The IP access list number.
<1300-2699>	The IP access list number (expanded range).
prefix-list	Use an IP prefix list to specify which next hop to match.
<prefix-listname>	The prefix list name.

Mode Route-map Configuration

Usage This command is valid for OSPF and RIP routes.

Examples To add entry 3 to the route map called rmap1, which will process routes whose next hop matches the ACL called mylist, use the commands:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# match ip next-hop mylist
```

To add entry `3` to the route map called `mymap`, which will process routes whose next hop matches the prefix list called `list1`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map mymap permit 3
awplus(config-route-map)# match ip next-hop prefix-list list1
```

Related Commands

access-list (extended numbered)
access-list (standard numbered)
ip prefix-list
route-map
show ip access-list
show ip prefix-list
show route-map

match ipv6 address

Use this command to specify the match address of route.

Use the **no** variant of this command to remove the `match ipv6 address` entry.

Syntax

```
match ipv6 address {<access-list-name>|prefix-list <prefix-listname>}
no match ipv6 address [<access-list-name>|prefix-list <prefix-
    listname>]
```

Parameter	Description
<access-list-name>	The name of the IPv6 access list that specifies criteria for the addresses to match. Valid only with RIPng.
<prefix-listname>	The name of the IPv6 prefix list that specifies criteria for the addresses to be matched. Valid only with RIPng.

Mode Route-map Configuration

Usage The **match ipv6 address <access-list>** command specifies the IPv6 address to be matched. If there is a match for the specified IPv6 address, and `permit` is specified, the route is redistributed or controlled as specified by the set action. If the match criteria are met, and `deny` is specified, the route is not redistributed or controlled. If the match criteria are not met, the route is neither accepted nor forwarded, irrespective of `permit` or `deny` specifications.

The **match ipv6 address prefix-list** command specifies the entries of prefix-lists to be matched. If there is a match for the specified prefix-list entries, and `permit` is specified, the route is redistributed or controlled as specified by the set action. If the match criteria are met, and `deny` is specified, the route is not redistributed or controlled. If the match criteria are not met, the route is neither accepted nor forwarded, irrespective of `permit` or `deny` specifications.

Examples

```
awplus# configure terminal
awplus(config)# route-map rmap1 deny 1
awplus(config-route-map)# match ipv6 address rmap1

awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# match ipv6 address prefix-list mylist
```

match metric

Use this command to add a metric match clause to a route map entry. Specify the metric value to match.

A route matches the route map if its metric matches the route map's metric.

Each entry of a route map can only match against one metric value in one metric match clause. If the route map entry already has a metric match clause, entering this command replaces that match clause with the new clause.

Use the **no** variant of this command to remove the metric match clause from the route map entry.

Syntax

```
match metric <metric>
no match metric [<metric>]
```

Parameter	Description
<metric>	*<0-4294967295>* Specifies the metric value.

Mode Route-map Configuration

Usage This command is valid for OSPF and RIP routes.

Example To stop entry `3` of the route map called `myroute` from processing routes with a metric of `888999`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map myroute permit 3
awplus(config-route-map)# no match metric 888999
```

Related Commands

route-map
set metric
show route-map

match route-type

Use this command to add an external route-type match clause to a route map entry. Specify whether to match OSPF type-1 external routes or OSPF type-2 external routes.

An OSPF route matches the route map if its route type matches the route map's route type.

Each entry of a route map can only match against one route type in one match clause. If the route map entry already has a route type match clause, entering this command replaces that match clause with the new clause.

Use the **no** variant of this command to remove the route type match clause from the route map entry.

Syntax

```
match route-type external {type-1|type-2}
no match route-type external [type-1|type-2]
```

Parameter	Description
`type-1`	OSPF type-1 external routes.
`type-2`	OSPF type-2 external routes.

Mode Route-map Configuration

Usage Use the **match route-type external** command to match specific external route types. AS-external LSA is either Type-1 or Type-2. **external type-1** matches only Type 1 external routes, and **external type-2** matches only Type 2 external routes.

This command is valid for OSPF routes only.

Example To add entry `10` to the route map called `mymap1`, which will process type-1 external routes, use the commands:

```
awplus# configure terminal
awplus(config)# route-map mymap1 permit 10
awplus(config-route-map)# match route-type external type-1
```

Related Commands

match interface
match ip address
match ip next-hop
match tag
route-map
set metric-type
show route-map

match tag

Use this command to add a tag match clause to a route map entry. Specify the route tag value to match.

An OSPF route matches the route map if it has been tagged with the route map's tag value. Routes can be tagged through OSPF commands or through another route map's set clause.

Each entry of a route map can only match against one tag in one match clause. If the route map entry already has a tag match clause, entering this command replaces that match clause with the new clause.

Use the **no** variant of this command to remove the tag match clause from the route map entry.

Syntax

```
match tag <0-4294967295>
no match tag [<0-4294967295>]
```

Mode Route-map Configuration

Usage This command is valid for OSPF routes only.

Example To add entry 10 to the route map called `mymap1`, which will process routes that are tagged `100`, use the following commands:

```
awplus# configure terminal
awplus(config)# route-map mymap1 permit 10
awplus(config-route-map)# match tag 100
```

Related Commands

match interface
match ip address
match ip next-hop
match route-type
route-map
set tag
show route-map

route-map

Use this command to configure a route map entry, and to specify whether the device will process or discard matching routes.

The switch uses a name to identify the route map, and a sequence number to identify each entry in the route map.

The **route-map** command puts you into route-map configuration mode. In this mode, you can use the following:

- one or more of the **match** commands to create match clauses. These specify what routes match the entry.
- one or more of the **set** commands to create set clauses. These change the attributes of matching routes.

Use the **no** variant of this command to delete a route map or to delete an entry from a route map.

Syntax

```
route-map <mapname> {deny|permit} <seq>
no route-map <mapname>
no route-map <mapname> {deny|permit} <seq>
```

Parameter	Description
<mapname>	A name to identify the route map.
deny	The route map causes a routing process to discard matching routes.
permit	The route map causes a routing process to use matching routes.
<seq>	<1-65535> The sequence number of the entry. You can use this parameter to control the order of entries in this route map.

Mode Global Configuration

Usage Route maps allow you to control and modify routing information by filtering routes and setting route attributes. You can apply route maps when the device:

- redistributes routes from one routing protocol into another
- redistributes static routes into routing protocols

When a routing protocol passes a route through a route map, it checks the entries in order of their sequence numbers, starting with the lowest numbered entry.

If it finds a match on a route map with an action of permit, then it applies any set clauses and accepts the route. Having found a match, the route is not compared against any further entries of the route map.

If it finds a match on a route map with an action of deny, it will discard the matching route.

If it does not find a match, it discards the route. This means that route maps end with an implicit deny entry. To permit all non-matching routes, end your route map with an entry that has an action of **permit** and no match clause.

Examples

To enter route-map mode for entry 2 of the route map called `route1`, and then add a match and set clause to it, use the commands:

```
awplus# configure terminal
awplus(config)# route-map route1 permit 2
awplus(config-route-map)# match interface vlan2
awplus(config-route-map)# set metric 20
```

Note how the prompt changes when you go into route map configuration mode.

To make the device process non-matching routes instead of discarding them, add a command like the following one:

```
awplus(config)# route-map route1 permit 100
```

Related Commands

show route-map

For OSPF:
distribute-list (OSPF)
default-information originate (OSPF)
redistribute (OSPF)

For RIP:
redistribute (RIP)

set ip next-hop (route map)

Use this command to add a next-hop set clause to a route map entry.

When a route matches the route map entry, the device sets the route's next hop to the specified IP address.

Use the **no** variant of this command to remove the set clause.

Syntax

```
set ip next-hop <ip-address>
no set ip next-hop [<ip-address>]
```

Parameter	Description
<ip-address>	The IP address of the next hop, entered in the form A.B.C.D.

Mode Route-map Configuration

Usage Use this command to set the next-hop IP address to the routes.

This command is valid for OSPF and RIP routes.

Example To use entry 3 of the route map called `mymap` to give matching routes a next hop of 10.10.0.67, use the commands:

```
awplus# configure terminal
awplus(config)# route-map mymap permit 3
awplus(config-route-map)# set ip next-hop 10.10.0.67
```

Related Commands
match ip next-hop
route-map
show route-map

set metric

Use this command to add a metric set clause to a route map entry.

When a route matches the route map entry, the device takes one of the following actions:

- changes the metric to the specified value, or
- adds or subtracts the specified value from the metric, if you specify **+** or **-** before the value (for example, to increase the metric by 2, enter +2)

Use the **no** variant of this command to remove the set clause.

Syntax

```
set metric {+<metric-value>|-<metric-value>|<metric-value>}
no set metric [+<metric-value>|-<metric-value> |<metric-value>]
```

Parameter	Description
+	Increase the metric by the specified amount.
–	Decrease the metric by the specified amount.
<metric-value>	<0-4294967295> The new metric value, or the amount by which to increase or decrease the existing value.

Default The default metric value for routes redistributed into OSPF and OSPFv3 is 20.

Mode Route-map Configuration

Usage This command is valid for OSPF and RIP routes.

Note that defining the OSPF metric in a route map supersedes the metric defined using a **redistribute (OSPF)** or a **redistribute (IPv6 OSPF)** command.

See the section **OSPF Metrics** in the **OSPF Introduction and Configuration** chapter for more information about OSPF metrics, and see the section **OSPFv3 Metrics** in the **OSPFv3 for IPv6 Introduction and Configuration** chapter for more information about OSPFv3 metrics.

Examples To use entry `3` of the route map called `rmap1` to give matching routes a metric of `600`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# set metric 600
```

To use entry `3` of the route map called `rmap1` to increase the metric of matching routes by `2`, use the commands:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# set metric +2
```

Related Commands

match metric
route-map
show route-map

set metric-type

Use this command to add a metric-type set clause to a route map entry.

When a route matches the route map entry, the device sets its route type to the specified value.

Use the **no** variant of this command to remove the set clause.

Syntax

```
set metric-type {type-1|type-2}
no set metric-type [type-1|type-2]
```

Parameter	Description
`type-1`	Redistribute matching routes into OSPF as type-1 external routes.
`type-2`	Redistribute matching routes into OSPF as type-2 external routes.

Mode Route-map Configuration

Usage This command is valid for OSPF routes only.

Example To use entry `3` of the route map called `rmap1` to redistribute matching routes into OSPF as type-1 external routes, use the commands:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# set metric-type 1
```

Related Commands

default-information originate (OSPF)
redistribute (OSPF)
match route-type
route-map
show route-map

set tag

Use this command to add a tag set clause to a route map entry.

When a route matches the route map entry, the device sets its tag to the specified value when it redistributes the route into OSPF.

Use the **no** variant of this command to remove the set clause.

Syntax

```
set tag <tag-value>
no set tag [<tag-value>]
```

Parameter	Description
<tag-value>	<0-4294967295> Value to tag matching routes with.

Mode Route-map Configuration

Usage This command is valid only when redistributing routes into OSPF.

Example To use entry 3 of the route map called `rmap1` to tag matching routes with the number 6, use the commands:

```
awplus# configure terminal
awplus(config)# route-map rmap1 permit 3
awplus(config-route-map)# set tag 6
```

Related Commands

default-information originate (OSPF)
redistribute (OSPF)
match tag
route-map
show route-map

show route-map

Use this command to display information about one or all route maps.

Syntax `show route-map <map-name>`

Parameter	Description
<map-name>	A name to identify the route map.

Mode User Exec and Privileged Exec

Example To display information about the route-map named `example-map`, use the command:

```
awplus# show route-map example-map
```

Output **Figure 46-1: Example output from the show route-map command**

```
route-map example-map, permit, sequence 1
  Match clauses:
    ip address prefix-list example-pref
  Set clauses:
    metric 100
route-map example-map, permit, sequence 200
  Match clauses:
  Set clauses:
```

Related Commands route-map

Part 4: Multicast Applications

- Chapter 47 Multicast Introduction and Commands
- Chapter 48 IGMP and IGMP Snooping Introduction
- Chapter 49 IGMP and IGMP Snooping Commands
- Chapter 50 PIM-SM Introduction and Configuration
- Chapter 51 PIM-SM Commands
- Chapter 52 PIM-SMv6 Introduction and Configuration
- Chapter 53 PIM-SMv6 Commands
- Chapter 54 PIM-DM Introduction and Configuration
- Chapter 55 PIM-DM Commands
- Chapter 56 MLD and MLD Snooping Introduction and Commands

Chapter 47: Multicast Introduction and Commands

Multicast Introduction 47.2
Multicast Groups 47.3
Components in a Multicast Network 47.3
Command List 47.5
clear ip mroute 47.6
clear ip mroute statistics 47.7
clear ipv6 mroute 47.8
clear ipv6 mroute statistics 47.9
debug nsm mcast 47.10
debug nsm mcast6 ..... 47.11
ip mroute 47.12
ip multicast forward-first-packet 47.14
ip multicast route 47.15
ip multicast route-limit 47.17
ip multicast wrong-vif-suppression 47.18
ip multicast-routing 47.19
ipv6 multicast route 47.20
ipv6 multicast route-limit 47.23
ipv6 multicast-routing 47.24
multicast 47.25
show ip mroute 47.26
show ip mvif 47.28
show ip rpf 47.28
show ipv6 mroute 47.29
show ipv6 mif 47.31

Multicast Introduction

Multicasting is a technique developed to send packets from one location in a network to many other locations without any unnecessary packet duplication. In multicasting, one packet is sent from a source and is replicated as needed in the network to reach as many end-users as necessary.

Multicasting is different from broadcasting; while broadcast packets are sent to every possible receiver, multicast packets need only be forwarded to receivers that want them. The benefit of this technique is bandwidth conservation - it is the most economical technique for sending a packet stream to many locations simultaneously.

The IPv4 and IPv6 addressing for multicast packets works differently from unicast and broadcast packets. A multicast stream sends packets out with a destination IPv4 or IPv6 address that identifies a specific multicast group. It does not at all specify an end host, like unicast; or a whole subnet, like broadcast.

This makes multicasting a connectionless process. The server simply sends out its multicast UDP packets, with no idea who will receive them, or whether they are successfully received. In an IPv4 network, IGMP allows hosts to tell the network that they wish to receive a multicast stream, using the Internet Group Management Protocol (IGMP). This is a Layer 3 protocol; however Layer 2 switches can also conserve bandwidth within their LAN by using IGMP snooping to track which hosts require the data stream. For more information about IGMP and IGMP snooping, see **Chapter 48, IGMP and IGMP Snooping Introduction**.

Similar to IGMP for IPv4, Multicast Listener Discovery (MLD) for IPv6 is used to exchange membership status information between IPv6 routers that support multicasting and members of multicast groups on a network segment. Host membership in a multicast group is reported by individual member hosts, and membership status is periodically polled by multicast routers. For an overview of MLD and MLD command descriptions, see **Chapter 56, MLD and MLD Snooping Introduction and Commands**.

Note

IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Note

The IPv6 Multicast addresses shown can be derived from IPv6 unicast prefixes as per RFC 3306.The IPv6 unicast prefix reserved for documentation is 2001:0db8::/32 as per RFC 3849. Using the base /32 prefix the IPv6 multicast prefix for 2001:0db8::/32 is ff3x:20:2001:0db8::/64. Where an RP address is 2001:0db8::1 the embedded RP multicast prefix is ff7x:120:2001:0db8::/96. For ASM (Any-Source Multicast) the IPV6 multicast addresses allocated for documentation purposes are ff0x::0db8:0:0/96 as per RFC 6676. This is a /96 prefix so that it can be used with group IDs as per RFC 3307. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

Note

The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

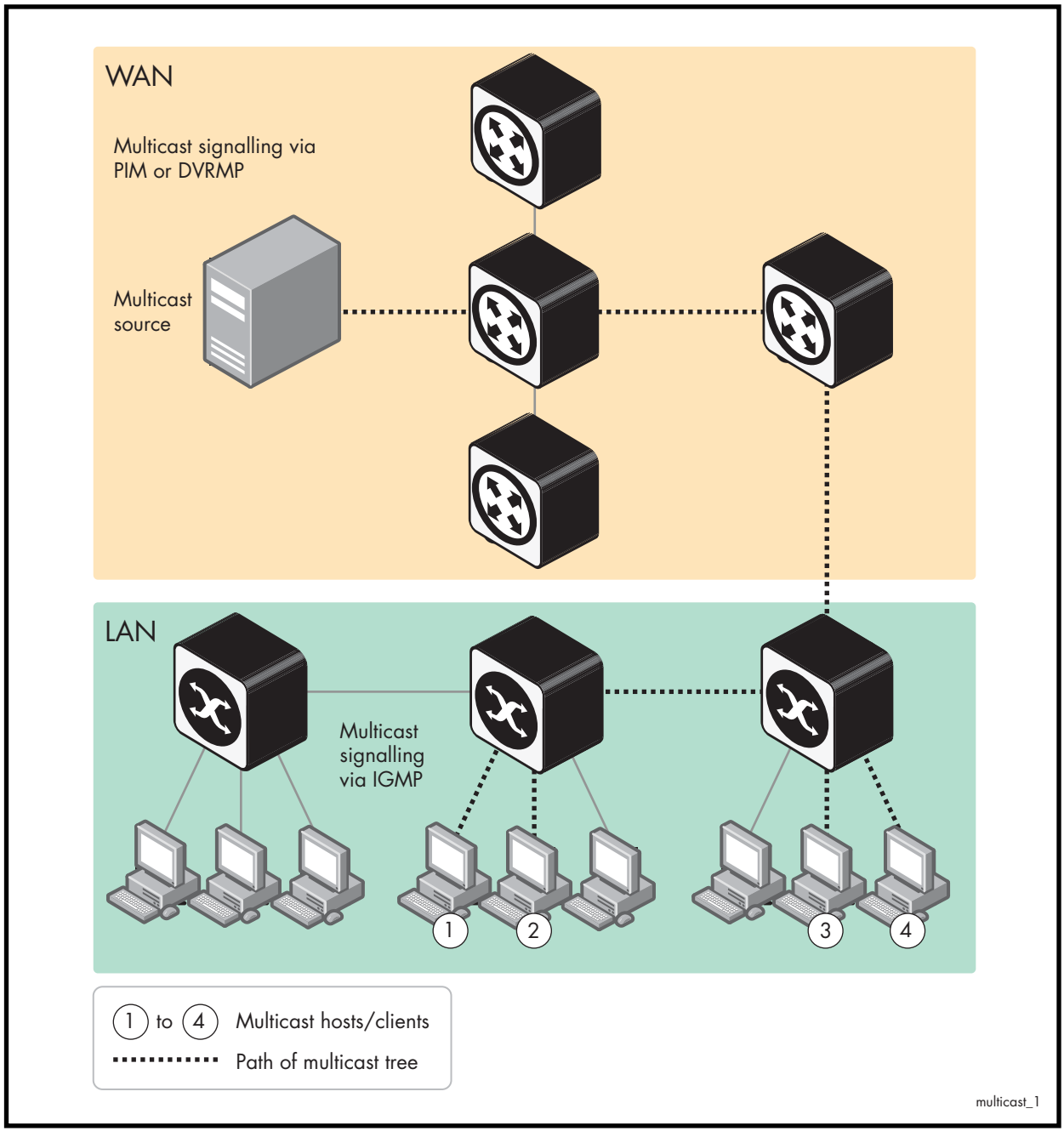
Multicast Groups

The concept of a group is crucial to multicasting. A group is the set of hosts that wish to receive a particular multicast stream, and is identified by a multicast IP address at Layer 3 and matching multicast MAC address at Layer 2. The multicast sender transmits the stream to the group address, and only members of the group can receive the multicast data.

Components in a Multicast Network

There are several protocols and roles required in a multicast network, as shown in **Figure 47-1 on page 47.3**. This section describes the end-to-end process of transporting multicast data through a network.

Figure 47-1: Components in a multicast network



At the two ends of a multicast data transmission are:

- **the source:** this is typically a server or video encoder. It sends the stream of multicast data out through its network interface. It is unaware of where the recipients of the stream are, or if there are any recipients.
- **the recipients:** these are storage or display devices, such as PCs, set-top boxes, or security video archivers. The recipients signal their desire to receive a particular multicast stream by sending out IGMP messages requesting the stream.

The role of the network in-between is to deliver the multicast stream to the recipients as efficiently as possible. The devices achieve this by exchanging signaling information between themselves in order to establish a forwarding path along which the multicast stream will flow. Each node informs the next node up the chain that it needs to receive the multicast stream. Once this series of requests reaches the router nearest the multicast source, then that router will start to forward the stream. All the nodes between the source and the recipients are ready to forward the stream, due to their having received the signaled requests. In this way, the stream is efficiently forwarded right through to the recipients.

The type of signaling that the network uses falls into two categories:

- in the local area network where the recipients are located, the signaling consists of the exchange of IGMP or MLD packets.
- as soon as the signaling needs to leave the VLAN containing the recipients and cross into other VLANs and subnets, then a Layer 3 multicasting protocol like PIM is used between the routers in the Layer 3 network.

In every Layer 2 multicast network, there needs to be a device that is sending IGMP or MLD queries into the network. This is essential to maintain multicast flows once they have been established (see **"Staying in the Multicast Group (Query Message)" on page 48.4** for more information). Typically, the device that is configured to send the queries is the router that is the gateway from the local network into a Layer 3 network.

Command List

This chapter provides an alphabetical reference of multicast commands common to PIM Sparse Mode and PIM Dense Mode. See also **Chapter 56, MLD and MLD Snooping Introduction and Commands**, **Chapter 55, PIM-DM Commands**, **Chapter 51, PIM-SM Commands** and **Chapter 49, IGMP and IGMP Snooping Commands**.

Note

IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Note

The IPv6 Multicast addresses shown can be derived from IPv6 unicast prefixes as per RFC 3306.The IPv6 unicast prefix reserved for documentation is 2001:0db8::/32 as per RFC 3849. Using the base /32 prefix the IPv6 multicast prefix for 2001:0db8::/32 is ff3x:20:2001:0db8::/64. Where an RP address is 2001:0db8::1 the embedded RP multicast prefix is ff7x:120:2001:0db8::/96. For ASM (Any-Source Multicast) the IPV6 multicast addresses allocated for documentation purposes are ff0x::0db8:0:0/96 as per RFC 6676. This is a /96 prefix so that it can be used with group IDs as per RFC 3307. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

Note

The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

clear ip mroute

Use this command to delete entries from the IPv4 multicast routing table.

Note If you use this command, you should also use the clear ip igmp group command to clear IGMP group membership records.

Syntax

```
clear ip mroute {*|<ipv4-group-address> [<ipv4-source-address>]}
   [pim sparse-mode]
```

Parameter	Description
*	Deletes all multicast routes.
<ipv4-group-address>	Group IPv4 address, in dotted decimal notation in the format A.B.C.D.
<ipv4-source-address>	Source IPv4 address, in dotted decimal notation in the format A.B.C.D.
pim sparse-mode	Clear specified IPv4 multicast route(s) for PIM Sparse Mode only.

Mode Privileged Exec

Usage When this command is used, the Multicast Routing Information Base (MRIB) clears the IPv4 multicast route entries in its IPv4 multicast route table, and removes the entries from the multicast forwarder. The MRIB sends a "clear" message to the multicast protocols. Each multicast protocol has its own "clear" multicast route command. The protocol-specific "clear" command clears multicast routes from PIM Sparse Mode, and also clears the routes from the MRIB.

Examples

```
awplus# clear ip mroute 225.1.1.1 192.168.3.3
```

```
awplus# clear ip mroute *
```

Related Commands ip multicast route
show ip mroute

clear ip mroute statistics

Use this command to delete multicast route statistics entries from the IP multicast routing table.

Syntax `clear ip mroute statistics {*|<ipv4-group-addr> [<ipv4-source-addr>]}`

Parameter	Description
*	All multicast route entries.
<ipv4-group-addr>	Group IPv4 address, in dotted decimal notation in the format `A.B.C.D`.
<ipv4-source-addr>	Source IPv4 address, in dotted decimal notation in the format `A.B.C.D`.

Mode Privileged Exec

Example

```
awplus# clear ip mroute statistics 225.1.1.2 192.168.4.4
```

```
awplus# clear ip mroute statistics *
```

clear ipv6 mroute

Use this command to delete one or more dynamically-added route entries from the IPv6 multicast routing table. You need to do this, for example, if you want to create a static route instead of an existing dynamic route.

Syntax

```
clear ipv6 mroute {*|<ipv6-group-address> [<ipv6-source-address>]}
```

Parameter	Description
*	Deletes all dynamically-learned IPv6 multicast routes.
<ipv6-group-address>	Group IPv6 address, in hexadecimal notation in the format `X.X::X.X`.
<ipv6-source-address>	Source IPv6 address, in hexadecimal notation in the format `X.X::X.X`.

Mode

Privileged Exec

Usage

When this command is used, the Multicast Routing Information Base (MRIB) clears the relevant IPv6 multicast route entries in its IPv6 multicast route table, and removes the entries from the multicast forwarder. The MRIB sends a “clear” message to the multicast protocols. Each multicast protocol has its own “clear” multicast route command.

This command does not remove static routes from the routing table or the configuration. To remove static routes, use the `no` parameter of the command **ipv6 multicast route**.

Example

```
awplus# clear ipv6 mroute 2001::2 ff08::1
```

Related Commands

ipv6 multicast route
show ipv6 mroute

clear ipv6 mroute statistics

Use this command to delete multicast route statistics entries from the IPv6 multicast routing table.

Note Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Syntax

```
clear ipv6 mroute statistics {*|<ipv6-group-address>
   [<ipv6-source-address>]}
```

Parameter	Description
*	All multicast route entries.
<ipv6-group-addr>	Group IPv6 address, in hexadecimal notation in the format `X.X::X.X`.
<ipv6-source-addr>	Source IPv6 address, in hexadecimal notation in the format `X.X::X.X`.

Mode Privileged Exec

Examples

```
awplus# clear ipv6 mroute statistics 2001::2 ff08::1
```

```
awplus# clear ipv6 mroute statistics *
```

debug nsm mcast

Use this command to debug IPv4 events in the Multicast Routing Information Base (MRIB).

Syntax

```
debug nsm mcast {all|fib-msg|mrt|mtrace|mtrace-detail|register|stats|
   vif}
```

Parameter	Description
`all`	All IPv4 multicast debugging.
`fib-msg`	Forwarding Information Base (FIB) messages.
`mrt`	Multicast routes.
`mtrace`	Multicast traceroute.
`mtrace-detail`	Multicast traceroute detailed debugging.
`register`	Multicast PIM register messages.
`stats`	Multicast statistics.
`vif`	Multicast interface.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# debug nsm mcast all

awplus# configure terminal
awplus(config)# debug nsm mcast fib-msg

awplus# configure terminal
awplus(config)# debug nsm mcast mrt

awplus# configure terminal
awplus(config)# debug nsm mcast mtrace

awplus# configure terminal
awplus(config)# debug nsm mcast mtrace-detail

awplus# configure terminal
awplus(config)# debug nsm mcast register

awplus# configure terminal
awplus(config)# debug nsm mcast stat
```

```
awplus# configure terminal
awplus(config)# debug nsm mcast vif
```

debug nsm mcast6

Use this command to debug IPv6 events in the Multicast Routing Information Base (MRIB).

Syntax

```
debug nsm mcast6 {all|fib-msg|mrt|mtrace|mtrace-detail|register|
   stats|vif}
```

Parameter	Description
`all`	All IPv4 multicast debugging.
`fib-msg`	Forwarding Information Base (FIB) messages.
`mif`	Multicast interfaces.
`mrt`	Multicast routes.
`register`	Multicast PIM register messages.
`stats`	Multicast statistics.

Mode Privileged Exec and Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# debug nsm mcast6 all

awplus# configure terminal
awplus(config)# debug nsm mcast6 fib-msg

awplus# configure terminal
awplus(config)# debug nsm mcast6 mif

awplus# configure terminal
awplus(config)# debug nsm mcast6 mrt

awplus# configure terminal
awplus(config)# debug nsm mcast6 register

awplus# configure terminal
awplus(config)# debug nsm mcast6 stats
```

ip mroute

Use this command to inform multicast of the RPF (Reverse Path Forwarding) route to a given IPv4 multicast source.

Use the **no** variant of this command to delete a route to an IPv4 multicast source.

Syntax

```
ip mroute <ipv4-source-address/mask-length> [bgp|ospf|rip|static]
   <rpf-address> [<admin-distance>]

no ip mroute <ipv4-source-address/mask-length> [bgp|ospf|rip|static]
```

Parameter	Description
<ipv4-source-address/mask-length>	A multicast source IPv4 address and mask length, in dotted decimal notation in the format `A.B.C.D/M`.
`ospf`	OSPF unicast routing protocol.
`rip`	RIP unicast routing protocol.
`static`	Specifies a static route.
<rpf-address>	`A.B.C.D` The closest known address on the multicast route back to the specified source. This host IPv4 address can be within a directly connected subnet or within a remote subnet. In the case that the address is in a remote subnet, a lookup is done from the unicast route table to find the next hop address on the path to this host.
<admin-distance>	The administrative distance. Use this to determine whether the RPF lookup selects the unicast or multicast route. Lower distances have preference. If the multicast static route has the same distance as the other RPF sources, the multicast static route takes precedence. The default is `0` and the range available is `0-255`.

Mode Global Configuration

Usage Typically, when a Layer 3 multicast routing protocol is determining the RPF (Reverse Path Forwarding) interface for the path to an IPv4 multicast source, it uses the unicast route table to find the best path to the source. However, in some networks a deliberate choice is made to send multicast via different paths to those used for unicast. In this case, the interface via which a multicast stream from a given source enters a router may not be the same as the interface that connects to the best unicast route to that source.

This command enables the user to statically configure the switch with “multicast routes” back to given sources. When performing the RPF check on a stream from a given IPv4 source, the multicast routing protocol will look at these static entries as well as looking into the unicast routing table. The route with the lowest administrative distance - whether a static “multicast route” or a route from the unicast route table - will be chosen as the RPF route to the source.

Note that in this context the term “multicast route” does not imply a route via which the current router will forward multicast; instead it refers to the route the multicast will have traversed in order to arrive at the current router.

Examples

The following example creates a static multicast IPv4 route back to the sources in the `10.10.3.0/24` subnet. The multicast route is via the host `192.168.2.3`, and has an administrative distance of `2`:

```
awplus# configure terminal
awplus(config)# ip mroute 10.10.3.0/24 static 2 192.168.2.3 2
```

The following example creates a static multicast IPv4 route back to the sources in the `192.168.3.0/24` subnet. The multicast route is via the host `10.10.10.50`. The administrative distance on this route has the default value of `0`:

```
awplus# configure terminal
awplus(config)# ip mroute 192.168.3.0/24 10.10.10.50
```

Validation Commands

show ip rpf

ip multicast forward-first-packet

Use this command to enable multicast to forward the first multicast packets coming to the device.

Use the **no** variant of this command to disable this feature.

Syntax

```
ip multicast forward-first-packet
no ip multicast forward-first-packet
```

Default By default, this feature is disabled.

Mode Global Configuration

Usage If this command is enabled, the device will forward the first packets in a multicast stream that create the multicast route, possibly causing degradation in the quality of the multicast stream, such as the pixelation of video and audio data.

Note If you use this command, ensure that the ip igmp snooping command is enabled, the default setting, otherwise the device will not process the first packets of the multicast stream correctly.

The device will forward the first multicast packets to all interfaces which are on the same VLAN as those which asked for this multicast group.

Examples To enable the forwarding of the first multicast packets, use the following commands:

```
awplus# configure terminal
awplus(config)# ip multicast forward-first-packet
```

To disable the forwarding of the first multicast packets, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip multicast forward-first-packet
```

ip multicast route

Use this command to add an IPv4 static multicast route for a specific multicast source and group IPv4 address to the multicast Routing Information Base (RIB). This IPv4 multicast route is used to forward multicast traffic from a specific source and group ingressing on an upstream VLAN to a single or range of downstream VLANs.

Use the **no** variant of this command to either remove an IPv4 static multicast route set with this command or to remove a specific downstream VLAN interface from an IPv4 static multicast route for a specific multicast source and group IPv4 address.

Syntax

```
ip multicast route <ipv4-source-addr> <ipv4-group-addr>
    <upstream-vlan-id> [<downstream-vlan-id>]

no ip multicast route <ipv4-source-addr> <ipv4-group-addr>
    [<upstream-vlan-id> <downstream-vlan-id>]
```

Parameter	Description
<ipv4-source-addr>	Source IPv4 address, in dotted decimal notation in the format `A.B.C.D`.
<ipv4-group-addr>	Group IPv4 address, in dotted decimal notation in the format `A.B.C.D`.
<upstream-vlan-id>	Upstream VLAN interface on which the multicast packets ingress.
<downstream-vlan-id>	Downstream VLAN interface or range of VLAN interfaces to which the multicast packets are sent.

Default By default, this feature is disabled.

Mode Global Configuration

Usage Only one multicast route entry per IPv4 address and multicast group can be specified. Therefore, if one entry for a static multicast route is configured, PIM will not be able to update this multicast route in any way.

If a dynamic multicast route exists you cannot create a static multicast route with same source IPv4 address, group IPv4 address, upstream VLAN and downstream VLANs. An error message is displayed and logged. To add a new static multicast route, either wait for the dynamic multicast route to timeout or clear the dynamic multicast route with the **clear ip mroute** command.

To update an existing static multicast route entry with more or a new set of downstream VLANs, you must firstly remove the existing static multicast route and then add the new static multicast route with all downstream VLANs specified. If you attempt to update an existing static multicast route entry with an additional VLAN or VLANs an error message is displayed and logged.

To create a blackhole or null route where packets from a specified source and group address coming from an upstream VLAN are dropped rather than forwarded, do not specify the optional *<downstream-vlan-id>* parameter when entering this command.

To remove a specific downstream VLAN from an existing static multicast route entry, specify the VLAN you want to remove with the <*downstream-vlan-id*> parameter when entering the **no** variant of this command.

Examples To create a static multicast route for the multicast source IPv4 address `2.2.2.2` and group IPv4 address `224.9.10.11`, specifying the upstream VLAN interface as `vlan10` and the downstream VLAN interface as `vlan20`, use the following commands:

```
awplus# configure terminal
awplus(config)# ip multicast route 2.2.2.2 224.9.10.11 vlan10
vlan20
```

To create a blackhole route for the multicast source IPv4 address `2.2.2.2` and group IPv4 address `224.9.10.11`, specifying the upstream VLAN interface as `vlan10`, use the following commands:

```
awplus# configure terminal
awplus(config)# ip multicast route 2.2.2.2 224.9.10.11 vlan10
```

To create an IPv4 static multicast route for the multicast source IPv4 address `2.2.2.2` and group IP address `224.9.10.11`, specifying the upstream VLAN interface as `vlan10` and the downstream VLAN range as `vlan20-25`, use the following commands:

```
awplus# configure terminal
awplus(config)# ip multicast route 2.2.2.2 224.9.10.11 vlan10
vlan20-25
```

To remove the downstream VLAN `23` from the IPv4 static multicast route created with the above command, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip multicast route 2.2.2.2 224.9.10.11
vlan10 vlan23
```

To delete an IPv4 static multicast route for the multicast source IP address `2.2.2.2` and group IP address `224.9.10.11`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip multicast route 2.2.2.2 224.9.10.11
```

Related Commands **clear ip mroute**
show ip mroute

ip multicast route-limit

Use this command to limit the number of multicast routes that can be added to an IPv4 multicast routing table.

Use the **no** variant of this command to return the IPv4 route limit to the default.

Syntax

```
ip multicast route-limit <limit> [<threshold>]
no ip multicast route-limit
```

Parameter	Description
<limit>	*<1-2147483647>* Number of routes.
<threshold>	*<1-2147483647>* Threshold above which to generate a warning message. The mroute warning threshold must not exceed the mroute limit.

Default The default limit and threshold value is 2147483647.

Mode Global Configuration

Usage This command limits the number of multicast IPv4 routes (mroutes) that can be added to a router, and generates an error message when the limit is exceeded. If the threshold parameter is set, a threshold warning message is generated when this threshold is exceeded, and the message continues to occur until the number of mroutes reaches the limit set by the limit argument.

Examples

```
awplus# configure terminal
awplus(config)# ip multicast route-limit 34 24

awplus# configure terminal
awplus(config)# no ip multicast route-limit
```

ip multicast wrong-vif-suppression

Use this command to prevent unwanted multicast packets received on an unexpected VLAN being trapped to the CPU.

Use the **no** variant of this command to disable wrong VIF suppression.

Syntax

```
ip ip multicast wrong-vif-suppression
no ip multicast wrong-vif-suppression
```

Default By default, this feature is disabled.

Mode Global Configuration

Usage Use this command if there is excessive CPU load and multicast traffic is enabled. To confirm that VIF messages are being sent to the CPU use the **debug nsm mcast6** command.

Examples To enable the suppression of wrong VIF packets, use the following commands:

```
awplus# configure terminal
awplus(config)# ip multicast wrong-vif-suppression
```

To disable the suppression of wrong VIF packets, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip multicast wrong-vif-suppression
```

ip multicast-routing

Use this command to turn on/off IPv4 multicast routing on the router; when turned off the device does not perform multicast functions.

Use the **no** variant of this command to disable IPv4 multicast routing after enabling it. Note the default stated below.

Syntax

```
ip multicast-routing
no ip multicast-routing
```

Default By default, IPv4 multicast routing is off.

Mode Global Configuration

Usage When the **no** variant of this command is used, the Multicast Routing Information Base (MRIB) cleans up Multicast Routing Tables (MRT), stops IGMP operation, and stops relaying multicast forwarder events to multicast protocols.

When multicast routing is enabled, the MRIB starts processing any MRT addition/deletion requests, and any multicast forwarding events.

You must enable multicast routing before issuing other multicast commands.

Example

```
awplus# configure terminal
awplus(config)# ip multicast-routing
```

Validation Commands **show running-config**

ipv6 multicast route

Use this command to add an IPv6 static multicast route for a specific multicast source and group IPv6 address to the multicast Routing Information Base (RIB). This IPv6 multicast route is used to forward IPv6 multicast traffic from a specific source and group ingressing on an upstream VLAN to a single or range of downstream VLANs.

See detailed usage notes below to configure static multicast router ports when using static IPv6 multicast routes with EPSR, and the destination VLAN is an EPSR data VLAN.

Use the **no** variant of this command to either remove an IPv6 static multicast route set with this command or to remove a specific downstream VLAN interface from an IPv6 static multicast route for a specific IPv6 multicast source and group address.

Syntax

```
ipv6 multicast route <ipv6-source-addr> <ipv6-group-addr>
    <upstream-vlan-id> [<downstream-vlan-id>]

no ipv6 multicast route <ipv6-source-addr> <ipv6-group-addr>
    [<upstream-vlan-id> <downstream-vlan-id>]
```

Parameter	Description
<ipv6-source-addr>	Source IPv6 address, in dotted decimal notation in the format `X.X::X.X`.
<ipv6-group-addr>	Group IP address, in dotted decimal notation in the format `X.X::X.X`.
<upstream-vlan-id>	Upstream VLAN interface on which the multicast packets ingress.
<downstream-vlan-id>	Downstream VLAN interface or range of VLAN interfaces to which the multicast packets are sent.

Default By default, no static routes exist.

Mode Global Configuration

Usage Only one multicast route entry per IPv6 address and multicast group can be specified. Therefore, if one entry for an IPv6 static multicast route is configured, PIM will not be able to update this multicast route in any way.

If a dynamic multicast route exists, you cannot create a static multicast route with the same source IPv6 address and group IPv6 address. An error message is displayed and logged. To add a new static multicast route, either wait for the dynamic multicast route to time out or clear the dynamic multicast route with the **clear ipv6 mroute** command.

To update an existing IPv6 static multicast route entry with new or additional downstream VLANs, you must firstly remove the existing static multicast route and then add the new static multicast route with all downstream VLANs specified. If you attempt to update an existing static multicast route entry with an additional VLAN or VLANs an error message is displayed and logged.

To create a blackhole or null route where packets from a specified source and group address coming from an upstream VLAN are dropped rather than forwarded, do not specify the optional *<downstream-vlan-id>* parameter when entering this command.

To remove a specific downstream VLAN from an existing static multicast route entry, specify the VLAN you want to remove with the <*downstream-vlan-id*> parameter when entering the **no** variant of this command.

Note that if static IPv6 multicast routing is being used with EPSR and the destination VLAN is an EPSR data VLAN, then multicast router (mrouter) ports must be statically configured. This minimizes disruption for multicast traffic in the event of ring failure or restoration.

When configuring the EPSR data VLAN, statically configure mrouter ports so that the multicast router can be reached in either direction around the EPSR ring.

For example, if `port1.0.1` and `port1.0.14` are ports on an EPSR data VLAN `vlan101`, which is the destination for a static IPv6 multicast route, then configure both ports as multicast router (mrouter) ports as shown in the example commands listed below:

Output **Figure 47-2: Example ipv6 mld snooping mrouter commands when static IPv6 multicast routing is being used and the destination VLAN is an ESPR data VLAN:**

```
awplus>enable
awplus#configure terminal
awplus(config)#interface vlan101
awplus(config-if)#ipv6 mld snooping mrouter interface port1.0.1
awplus(config-if)#ipv6 mld snooping mrouter interface port1.0.14
```

See **ipv6 mld snooping mrouter** for a command description and command examples.

Examples To create an IPv6 static multicast route for the multicast source IPv6 address `2001::1` and group IPv6 address `ff08::1`, specifying the upstream VLAN interface as `vlan10` and the downstream VLAN interface as `vlan20`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 multicast route 2001::1 ff08::1 vlan10
vlan20
```

To create a blackhole route for the IPv6 multicast source IP address `2001::1` and group IP address `ff08::1`, specifying the upstream VLAN interface as `vlan10`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 multicast route 2001::1 ff08::1 vlan10
```

To create an IPv6 static multicast route for the multicast source IPv6 address `2001::1` and group IPv6 address `ff08::1`, specifying the upstream VLAN interface as `vlan10` and the downstream VLAN range as `vlan20-25`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 multicast route 2001::1 ff08::1 vlan10
vlan20-25
```

To remove the downstream VLAN `23` from the IPv6 static multicast route created with the above command, use the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 multicast route 2001::1 ff08::1 vlan10
vlan23
```

To delete an IPv6 static multicast route for the multicast source IPv6 address `2001::1` and group IPv6 address `ff08::1`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 multicast route 2001::1 ff08::1
```

Related Commands

clear ipv6 mroute
ipv6 mld snooping mrouter
show ipv6 mroute

ipv6 multicast route-limit

Use this command to limit the number of multicast routes that can be added to an IPv6 multicast routing table.

Use the **no** variant of this command to return the IPv6 route limit to the default.

Syntax

```
ipv6 multicast route-limit <limit> [<threshold>]
no ipv6 multicast route-limit
```

Parameter	Description
<limit>	*<1-2147483647>* Number of routes.
<threshold>	*<1-2147483647>* Threshold above which to generate a warning message. The mroute warning threshold must not exceed the mroute limit.

Default The default limit and threshold value is 2147483647.

Mode Global Configuration

Usage This command limits the number of multicast IPv6 routes (mroutes) that can be added to a router, and generates an error message when the limit is exceeded. If the threshold parameter is set, a threshold warning message is generated when this threshold is exceeded, and the message continues to occur until the number of mroutes reaches the limit set by the limit argument.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 multicast route-limit 34 24

awplus# configure terminal
awplus(config)# no ipv6 multicast route-limit
```

ipv6 multicast-routing

Use this command to turn on/off IPv6 multicast routing on the router; when turned off the device does not perform multicast functions.

Use the **no** variant of this command to disable IPv6 multicast routing after enabling it. Note the default stated below.

Syntax

```
ipv6 multicast-routing
no ipv6 multicast-routing
```

Default By default, IPv6 multicast routing is off.

Mode Global Configuration

Usage When the **no** variant of this command is used, the Multicast Routing Information Base (MRIB) cleans up Multicast Routing Tables (MRT, and stops relaying multicast forwarder events to multicast protocols.

When multicast routing is enabled, the MRIB starts processing any MRT addition/deletion requests, and any multicast forwarding events.

You must enable multicast routing before issuing other multicast commands.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 multicast-routing

awplus# configure terminal
awplus(config)# no ipv6 multicast-routing
```

Validation Commands **show running-config**

multicast

Use this command to enable a switch port to route multicast packets that ingress the port.

Use the **no** variant of this command to stop the switch port from routing multicast packets that ingress the port. Note that this does not affect Layer 2 forwarding of multicast packets. If you enter **no multicast** on a port, multicast packets received on that port will not be forwarded to other VLANs, but ports in the same VLANs as the receiving port will still receive the multicast packets.

Syntax

```
multicast
no multicast
```

Default By default, all switch ports route multicast packets.

Mode Interface Configuration

Examples

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# multicast

awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no multicast
```

Validation Commands **show running-config**

show ip mroute

Use this command to display the contents of the IPv4 multicast routing (mroute) table.

Syntax

```
show ip mroute [<ipv4-group-addr>] [<ipv4-source-addr>] [{dense|
   sparse}] [{count|summary}]
```

Parameter	Description
<ipv4-group-addr>	Group IPv4 address, in dotted decimal notation in the format `A.B.C.D`.
<ipv4-source-addr>	Source IPv4 address, in dotted decimal notation in the format `A.B.C.D`.
`dense`	Display dense IPv4 multicast routes.
`sparse`	Display sparse IPv4 multicast routes.
`count`	Display the route and packet count from the IPv4 multicast routing (mroute) table.
`summary`	Display the contents of the IPv4 multicast routing (mroute) table in an abbreviated form.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip mroute 10.10.3.34 224.1.4.3
```

```
awplus# show ip mroute 10.10.5.24 225.2.2.2 count
```

```
awplus# show ip mroute 10.10.1.34 summary
```

Output The following is a sample output of this command displaying the IPv4 multicast routing table, with and without specifying the group and source IPv4 address:

Figure 47-3: Example output from the show ip mroute command

```
awplus# show ip mroute
IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder
installed
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(10.10.1.52, 224.0.1.3), uptime 00:00:31, stat expires 00:02:59
Owner PIM-SM, Flags: TF
  Incoming interface: vlan2
  Outgoing interface list:
   vlan3 (1)
```

Figure 47-4: Example output from the show ip mroute command with the source and group IPv4 address specified

```
awplus# show ip mroute 10.10.1.52 224.0.1.3

IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder
installed
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(10.10.1.52, 224.0.1.3), uptime 00:03:24, stat expires 00:01:28
Owner PIM-SM, Flags: TF
  Incoming interface: vlan2
  Outgoing interface list:
    vlan3 (1)
```

The following is a sample output of this command displaying the packet count from the IPv4 multicast routing table:

Figure 47-5: Example output from the show ip mroute count command

```
awplus# show ip mroute count
IP Multicast Statistics
Total 1 routes using 132 bytes memory
Route limit/Route threshold: 2147483647/2147483647
Total NOCACHE/WRONGVIF/WHOLEPKT recv from fwd: 1/0/0
Total NOCACHE/WRONGVIF/WHOLEPKT sent to clients: 1/0/0
Immediate/Timed stat updates sent to clients: 0/0
Reg ACK recv/Reg NACK recv/Reg pkt sent: 0/0/0
Next stats poll: 00:01:10

Forwarding Counts: Pkt count/Byte count, Other Counts: Wrong If
pkts
Fwd msg counts: WRONGVIF/WHOLEPKT recv
Client msg counts: WRONGVIF/WHOLEPKT/Imm Stat/Timed Stat sent
Reg pkt counts: Reg ACK recv/Reg NACK recv/Reg pkt sent

(10.10.1.52, 224.0.1.3), Forwarding: 2/19456, Other: 0
  Fwd msg: 0/0, Client msg: 0/0/0/0,  Reg: 0/0/0
```

The following is a sample output for this command displaying the IPv4 multicast routing table in an abbreviated form:

Figure 47-6: Example output from the show ip mroute summary command

```
awplus# show ip mroute summary

IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder
installed
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(10.10.1.52, 224.0.1.3), 00:01:32/00:03:20, PIM-SM, Flags: TF
```

show ip mvif

Use this command to display the contents of the IPv4 Multicast Routing Information Base (MRIB) VIF table.

Syntax `show ip mvif [<interface>]`

Parameter	Description
<interface>	The interface to display information about.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip mvif vlan2
```

Output **Figure 47-7: Example output from the show ip mvif command**

```
Interface    Vif   Owner    TTL  Local            Remote         Uptime
             Idx   Module         Address          Address
vlan2        0     PIM-SM   1    192.168.1.53     0.0.0.0        00:04:26
Register     1               1    192.168.1.53     0.0.0.0        00:04:26
vlan3        2     PIM-SM   1    192.168.10.53    0.0.0.0        00:04:25
```

Figure 47-8: Example output from the show ip mvif command with the interface parameter vlan2 specified

```
Interface   Vif   Owner    TTL  Local            Remote           Uptime
            Idx   Module         Address          Address
vlan2       0     PIM-SM   1    192.168.1.53     0.0.0.0          00:05:17
```

show ip rpf

Use this command to display Reverse Path Forwarding (RPF) information for the specified IPv4 source address.

Syntax `show ip rpf <source-addr>`

Parameter	Description
<ipv4-source-addr>	Source IPv4 address, in dotted decimal notation in the format A.B.C.D.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip rpf 10.10.10.50
```

show ipv6 mroute

Use this command to display the contents of the IPv6 multicast routing (mroute) table.

Syntax

```
show ipv6 mroute [<ipv6-group-addr>] [<ipv6-source-addr>] [{count|
   summary}]
```

Parameter	Description
<ipv6-group-addr>	Group IPv6 address, in hexadecimal notation in the format X.X::X.X.
<ipv6-source-addr>	Source IPv6 address, in hexadecimal notation in the format X.X::X.X.
count	Display the route and packet count from the IPv6 multicast routing (mroute) table.
summary	Display the contents of the IPv6 multicast routing (mroute) table in an abbreviated form.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ipv6 mroute

awplus# show ipv6 mroute count

awplus# show ipv6 mroute summary

awplus# show ipv6 mroute 2001::2 ff08::1 count

awplus# show ipv6 mroute 2001::2 ff08::1


awplus# show ipv6 mroute 2001::2 summary
```

Output The following is a sample output of this command displaying the IPv6 multicast routing table for a single static IPv6 Multicast route:

Figure 47-9: Example output from the show ipv6 mroute command

```
awplus#show ipv6 mroute
IPv6 Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder
installed
Timers: Uptime/Stat Expiry
Interface State: Interface
(2001::2, ff08::1), uptime 03:18:38
Owner IMI, Flags: F
  Incoming interface: vlan2
  Outgoing interface list:
    vlan3
```

The following is a sample output of this command displaying the IPv6 multicast routing count table for a single static IPv6 Multicast route:

Figure 47-10: Example output from the show ipv6 mroute count command

```
awplus#show ipv6 mroute count

IPv6 Multicast Statistics
Total 1 routes using 152 bytes memory
Route limit/Route threshold: 1024/1024
Total NOCACHE/WRONGmif/WHOLEPKT recv from fwd: 6/0/0
Total NOCACHE/WRONGmif/WHOLEPKT sent to clients: 6/0/0
Immediate/Timed stat updates sent to clients: 0/0
Reg ACK recv/Reg NACK recv/Reg pkt sent: 0/0/0
Next stats poll: 00:01:14

Forwarding Counts: Pkt count/Byte count, Other Counts: Wrong If
pkts
Fwd msg counts: WRONGmif/WHOLEPKT recv
Client msg counts: WRONGmif/WHOLEPKT/Imm Stat/Timed Stat sent
Reg pkt counts: Reg ACK recv/Reg NACK recv/Reg pkt sent

(2001::2, ff08::1), Forwarding: 0/0, Other: 0
  Fwd msg: 0/0, Client msg: 0/0/0/0,  Reg: 0/0/0
```

The following is a sample output of this command displaying the IPv6 multicast routing summary table for a single static IPv6 Multicast route:

Figure 47-11: Example output from the show ipv6 mroute summary command

```
awplus#show ipv6 mroute summary
IPv6 Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder
installed
Timers: Uptime/Stat Expiry
Interface State: Interface

(2001::2, ff08::1), 03:20:28/-, IMI, Flags: F
```

show ipv6 mif

Use this command to display the contents of the IPv6 Multicast Routing Information Base (MRIB) MIF table.

Syntax `show ipv6 mif [<interface>]`

Parameter	Description
<interface>	The interface to display information about.

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 mif

awplus# show ipv6 mif vlan2
```

Output **Figure 47-12: Example output from the show ipv6 mif command**

```
awplus#show ipv6 mif
Interface    Mif   Owner             Uptime
             Idx   Module
vlan3        0     MLD/MLD Proxy-Service 03:28:48
vlan2        1     MLD/MLD Proxy-Service 03:28:48
vlan1        2     MLD/MLD Proxy-Service 03:28:48
```

Figure 47-13: Example output from the show ipv6 mif command with the interface parameter vlan2 specified

```
Interface    Mif   Owner    TTL   Remote            Uptime
             Idx   Module         Address
vlan2        0     PIM-SMv6   1    0.0.0.0           00:05:17
```

Chapter 48: IGMP and IGMP Snooping Introduction

Introduction 48.2
IGMP 48.2
Joining a Multicast Group (Membership Report) 48.3
Staying in the Multicast Group (Query Message) 48.4
Leaving the Multicast Group (Leave Message) 48.4
IGMP Snooping 48.5
How IGMP Snooping Operates 48.5
IGMP Snooping and Querier Configuration Example 48.6
Query Solicitation 48.9
How Query Solicitation Works 48.9
Query Solicitation Operation 48.9
Speeding up IGMP Convergence in a Non-Looped Topology 48.12
Enabling Query Solicitation on Multiple Switches in a Looped Topology 48.12

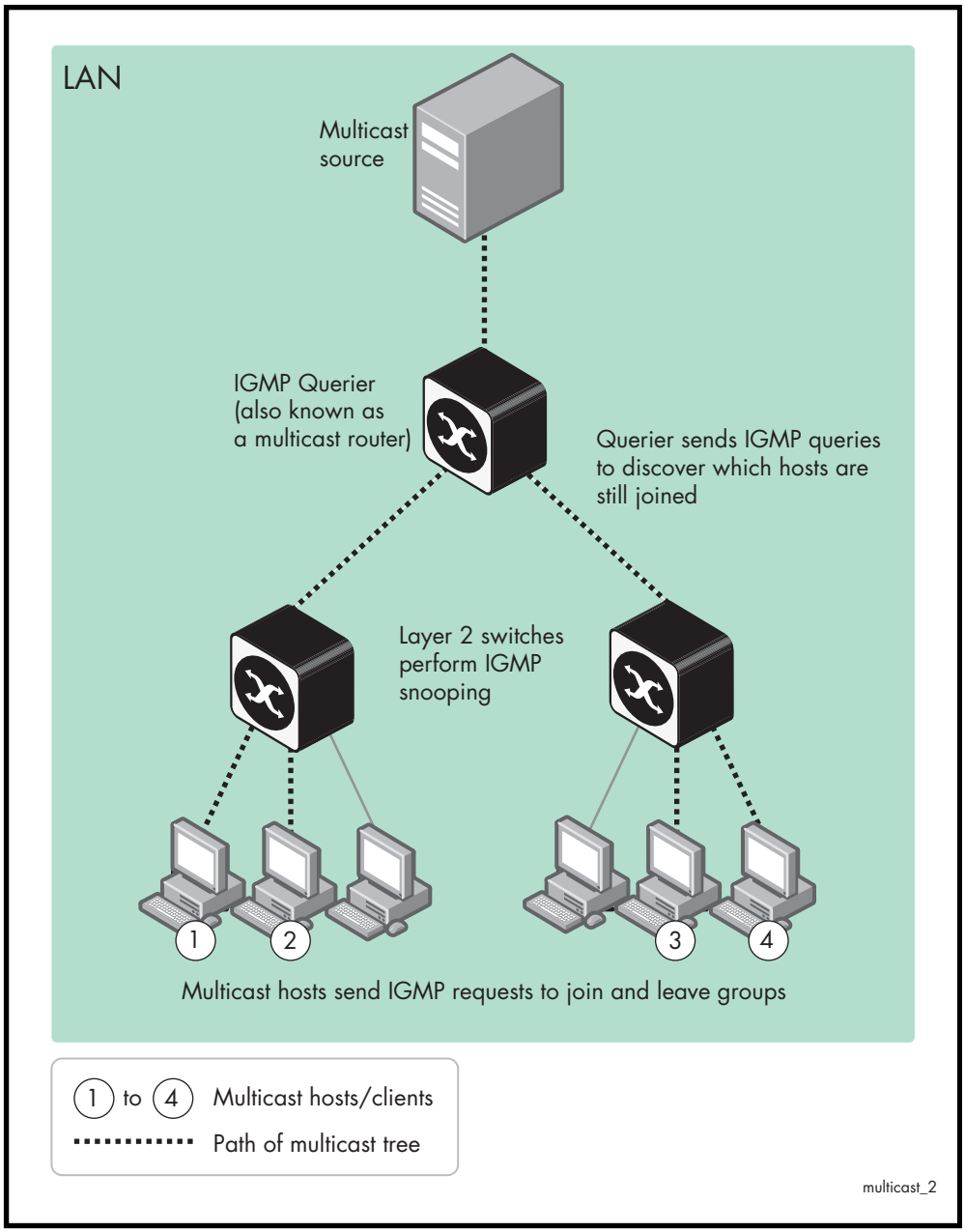
Introduction

This chapter provides information about Internet Group Management Protocol (IGMP), IGMP Snooping, and an introduction to the Query Solicitation feature when used with IGMP Snooping. To see details on the commands used in this example, or to see the outputs of the validation commands, refer to **Chapter 49, IGMP and IGMP Snooping Commands**. For a general overview of multicasting, see **Chapter 47, Multicast Introduction and Commands**.

IGMP

Internet Group Management Protocol (IGMP) is the protocol that hosts use to indicate that they are interested in receiving a particular multicast stream. An example of a multicast system within a single Layer 2 LAN is shown in **Figure 48-1**.

Figure 48-1: Multicast system within a single LAN



Joining a Multicast Group (Membership Report)

When a host wants to receive a stream, referred to as "joining a group", it sends out an IGMP packet containing the address of the group it wants to join. This packet is called an IGMP Membership report, often referred to as a "join packet". This packet is forwarded through the LAN to the local IGMP querier, which is typically a router. Once the querier has received an IGMP join message, it knows to forward the multicast stream to the host. If it is not already receiving the stream, it must tell the devices between itself and the multicast source, which may be some hops away from the querier, that it wishes to receive the stream. This might involve a process of using Layer 3 multicast protocols to signal across a WAN, or it might be as simple as receiving a stream from a locally connected multicast server.

Staying in the Multicast Group (Query Message)

The Query message is used by a querier to determine whether hosts are still interested in an IGMP group. At certain time intervals (the default is 125 seconds), the querier sends an IGMP query message onto the local LAN. The destination address of the query message is a special “all multicast groups” address. The purpose of this query is to ask “Are there any hosts on the LAN that wish to remain members of multicast groups?” After receiving an IGMP query, any host that wants to remain in a multicast group must send a new join packet for that group. If a host is a member of more than one group, then it sends a join message for each group it wants to remain a member of. The querier looks at the responses it receives to its query, and compares these to the list of multicast streams that it is currently registered to forward. If there are any items in that list for which it has not received query responses, it will stop forwarding those streams. Additionally, if it is receiving those streams through a Layer 3 network, it will send a Layer 3 routing protocol message upstream, asking to no longer receive that stream.

Leaving the Multicast Group (Leave Message)

How a host leaves a group depends on the IGMP version that it is using. Under IGMP version 1, when a host has finished with a data stream, the local querier continues to send the stream to the host until it sends out the next query message and receives no reply back from the host. IGMP version 2 introduced the Leave message. This allows a host to explicitly inform its querier that it wants to leave a particular multicast group. When the querier receives the Leave message, it sends out a group specific query asking whether any hosts still want to remain members of that specific group. If no hosts respond with join messages for that group, then the querier knows that there are no hosts on its LAN that are still members of that group. This means that for that specific group, it can ask to be pruned from the multicast tree. IGMP version 3 removed the Leave message. Instead a host leaves a group by sending a join message with no source specified.

IGMP Snooping

IGMP Snooping is a way for Layer 2 switches to reduce the amount of multicast traffic on a LAN. The AlliedWare Plus implementation of IGMP Snooping is compatible with networks running all IGMP versions.

Without IGMP Snooping, Layer 2 switches handle IP multicast traffic in the same manner as broadcast traffic and forward multicast frames received on one port to all other ports in the same VLAN. IGMP Snooping allows switches to monitor network traffic, and determine hosts to receive multicast traffic, by looking into IGMP packets to learn which attached hosts need to receive which multicast groups. This allows the switch to forward multicast traffic only out the appropriate ports. If it sees multiple reports sent for one group, it will forward only one of them.

How IGMP Snooping Operates

IGMP Snooping operates similarly to the multicast protocols. When a switch hears an IGMP report from a host for a given multicast group, the switch adds the host's port number to the list of ports that are listening to the multicast group. When the switch hears an IGMP leave, it removes the host's port from the list, after the completion of the leave process as described in **"Leaving the Multicast Group (Leave Message)" on page 48.4**. When there are no hosts listening to a group, the switch informs the local querier to stop sending that group's multicast stream.

IGMP Snooping allows query messages to be forwarded to all ports. The hosts that still require the stream respond to the queries by sending reports. The switch intercepts these. Depending on configuration settings, the switch may just forward the reports directly on to the querier, or it may proxy report on behalf of the group, only forwarding on one consolidated report for each group.

By default, IGMP Snooping is enabled both globally and on all VLANs.

Note

IGMP Snooping cannot be disabled on an interface if IGMP Snooping has already been disabled globally. IGMP Snooping can be disabled on both an interface and globally if disabled on the interface first and then disabled globally.

To disable IGMP Snooping either		
1.	`awplus#`	
	`configure terminal`	Enter Global Configuration mode.
2.	`awplus(config)#`	
	`no ip igmp snooping`	Disable IGMP Snooping globally.
or		
1.	`awplus#`	
	`configure terminal`	Enter Global Configuration mode.
2.	`awplus(config)#`	
	`interface <vlan-name>`	Enter Interface Configuration mode for a specific VLAN.
3.	`awplus(config-if)#`	
	`no ip igmp snooping`	Disable IGMP Snooping for a specific VLAN.

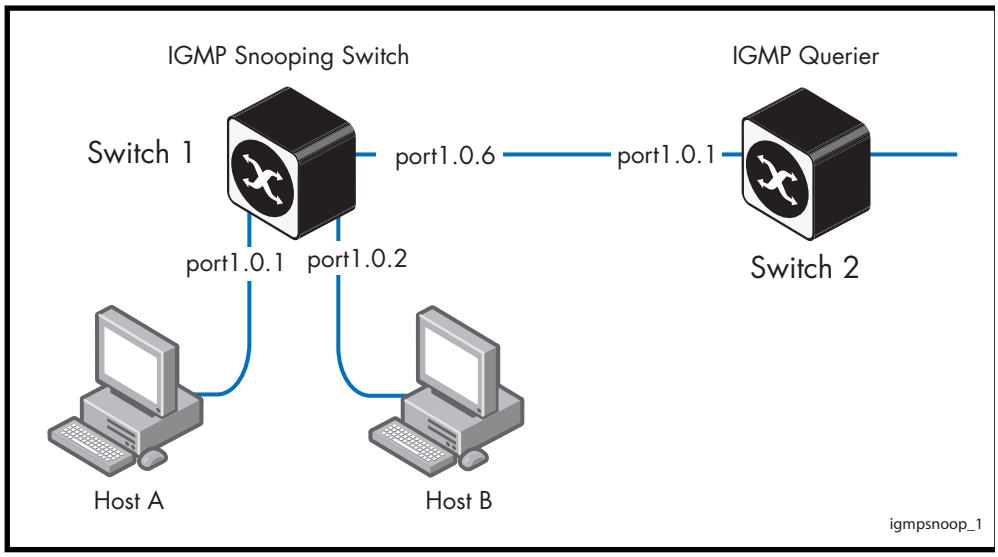
IGMP Snooping and Querier Configuration Example

This example describes the configuration of IGMP Snooping on an Allied Telesis managed Layer 3 switch (Switch 1) and the configuration of IGMP Querier (Switch 2). The interface port1.0.6 is configured as a multicast router port. Host A and Host B are both members of the same multicast group.

To enable IGMP Snooping on an interface:

- Enable IGMP Snooping globally, if necessary. IGMP Snooping is enabled by default.
- Enable IGMP Snooping on the desired interfaces, if necessary. IGMP Snooping is enabled on all interfaces by default.
- Statically configure ports that are connected to routers if necessary.

Figure 48-2: IGMP Snooping configuration example



As a result of this configuration:

- Membership reports are generated by hosts. The IGMP Snooping switch will forward the membership reports to its router port. Queries received by the IGMP Snooping switch from the IGMP Querier on port1.0.6 are forwarded by the IGMP Snooping switch.
- Because Host A and Host B are members of the same multicast group, the switch does not notify the IP IGMP routing device (IGMP Querier) when Host A leaves the group, because the group still has another member Host B remaining. When Host B also leaves the group, the switch forwards the leave message to the IP IGMP Querier.
- The addition of a static mrouter port is only required when there is no upstream IGMP querier or an upstream router does not send topology discovery or maintenance messages (like IGMP General Queries or OSPF Hello packets).
- In this example, the configuration of a static mrouter port on port1.0.6 is provided to illustrate the **ip igmp snooping mrouter** command. However, this command would probably not be necessary, since the switch should dynamically set port1.0.6 to be an mrouter port as it receives IGMP Queries arriving from the IGMP Querier attached to port1.0.6.

- In this example, it is not necessary to explicitly configure the switch to work with IGMPv2 or IGMPv3. When the IGMP version is not configured then the switch will work with both versions of IGMP.

Table 48-1: Configuring IGMP Snooping on Switch 1 and IGMP Querier on Switch 2

Step 1.	**Configure IGMP Snooping (Switch 1)**	
1.	`awplus#` `configure terminal`	Enter Global Configuration mode.
2.	`awplus(config)#` `ip igmp snooping`	Enable IGMP Snooping globally. Snooping is enabled by default. Use this command only if you have previously disabled it.
3.	`awplus(config)#` `interface vlan1`	Enter Interface Configuration mode for VLAN 1.
4.	`awplus(config-if)#` `ip igmp snooping mrouter interface port1.0.6`	Configure port1.0.6 as a multicast router port to the IGMP Querier.
5.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
Step 2.	**Validate the configuration**	
6.	`awplus#` `exit`	Return to Privileged Exec mode.
7.	`awplus#` `show ip igmp interface vlan1`	Display the state of IGMP Snooping for VLAN 1.
8.	`awplus#` `show ip igmp groups`	Display the multicast groups with receivers directly connected to the router.
9.	`awplus#` `show ip igmp snooping mrouter interface vlan1`	Display the multicast router ports, both static and dynamic, in VLAN 1.
Step 3.	**Configure IGMP Querier (Switch 2)**	
1.	`awplus#` `configure terminal`	Enter Global Configuration mode.
2.	`awplus(config)#` `interface vlan1`	Enter Interface Configuration mode for VLAN 1.
3.	`awplus(config-if)#` `ip igmp`	Enable IGMP on VLAN 1 and configure the switch as an IGMP Querier.
Step 4.	**Validate the configuration**	
4.	`awplus#` `exit`	Return to Privileged Exec mode.

Table 48-1: Configuring IGMP Snooping on Switch 1 and IGMP Querier on Switch 2

5.	`awplus#` `show ip igmp interface vlan1`	Display the state of IGMP Querier for VLAN 1.
6.	`awplus#` `show running-config`	Display the current dynamic configuration of Switch 2.

Query Solicitation

Query Solicitation minimizes the loss of multicast data after a topology change on networks that use EPSR or spanning tree (STP, RSTP, or MSTP) for loop protection. Without Query Solicitation, when the underlying link layer topology changes, multicast data flow can stop for up to several minutes, depending on which port goes down and how much of the IGMP query interval remained at the time of the topology change. Query Solicitation greatly reduces this disruption.

Query Solicitation operates without configuration in AlliedWare Plus™ switches running STP, RSTP, MSTP or EPSR. However, you may find it useful to manually enable Query Solicitation in loop-free networks running IGMP (see **Speeding up IGMP Convergence in a Non-Looped Topology**) and networks where not all switches support Query Solicitation (see **Enabling Query Solicitation on Multiple Switches in a Looped Topology**).

How Query Solicitation Works

Query Solicitation monitors STP, RSTP, MSTP and EPSR messages for topology changes. When it detects a change, it generates a special IGMP Leave message called a Query Solicit. The switch floods the Query Solicit message to all ports in every VLAN that Query Solicitation is enabled on. When the Querier receives the Query Solicit message, it sends out a General Query and waits for clients to respond with Membership Reports. These Reports update the snooping information throughout the network.

Query Solicit messages have a group address of 0.0.0.0.

Query Solicitation works by default (without you enabling it) on all VLANs on the root bridge in an STP instance and on all data VLANs on the master node in an EPSR instance. By default, the root bridge or master node always sends a Query Solicit message when any of the following events occur:

- an STP BPDU packet with the Topology Change (TC) flag arrives at the root bridge
- an STP port on a switch goes from a Discarding to Forwarding state
- the FDB gets flushed by EPSR

If necessary, you can make clients respond more quickly to the General Query by tuning the IGMP timers, especially the maximum response time advertised in IGMP queries using the **ip igmp query-max-response-time** command.

Query Solicitation Operation

When IGMP Snooping is enabled and EPSR or Spanning Tree changes the underlying link layer topology, this can interrupt multicast data flow for a significant length of time. This is because there is no way for switches in a network with interested clients to know where the traffic is available, due to the change in network topology. This change in network topology may take up to two IGMP Query intervals from the IGMP Querier, until the switches will know where to forward membership reports received by client hosts. During this time, those hosts will not receive multicast traffic.

Query solicitation prevents this by monitoring for any topology changes. When it detects a change, it generates a special IGMP Leave message known as a Query Solicit, and floods the Query Solicit message to all ports in every VLAN that query solicitation is enabled on. When the IGMP Querier receives the message, it responds by sending a General Query, which all IGMP listeners respond to. This refreshes snooped group membership information in the network.

Query solicitation reduces downtime to a negligible amount by triggering on topology changes. The generation of query solicitation messages in the network causes the IGMP Querier to send an IGMP Query immediately following a topology change. This enables the switches to know where to look for the traffic and thus to send reports to the correct switch upstream. This allows the multicast data traffic to be recovered instantly.

Query solicitation functions by default (without you enabling it) on all VLANs on the root bridge in an STP instance and on all data VLANs on the master node in an EPSR instance. By default, the root bridge or master node always sends a Query Solicit message when the topology changes.

If you have multiple STP or EPSR instances, query solicitation only sends Query Solicit messages on VLANs in the instance that experienced a topology change.

In switches other than the STP root bridge or EPSR master node, query solicitation is disabled by default, but you can enable it by using the **ip igmp snooping tcn query solicit** command.

If you enable query solicitation on a switch other than the STP root bridge or EPSR master node, both that switch and the root or master send a Query Solicit message.

Once the Querier receives the Query Solicit message, it sends out a General Query and waits for responses, which update the snooping information throughout the network.

The **ip igmp query-holdtime** command can be configured on the IGMP Querier. This command introduces a brief delay between when the IGMP Querier receives the query solicit, and when it sends out the general query. Although this slightly reduces the speed with which the network recovers from the topology change, its does guard against a DoS (Denial of Service) attack. Without this delay, a malign host sending a stream of query solicits could cause the IGMP Querier to flood the network with IGMP Queries.

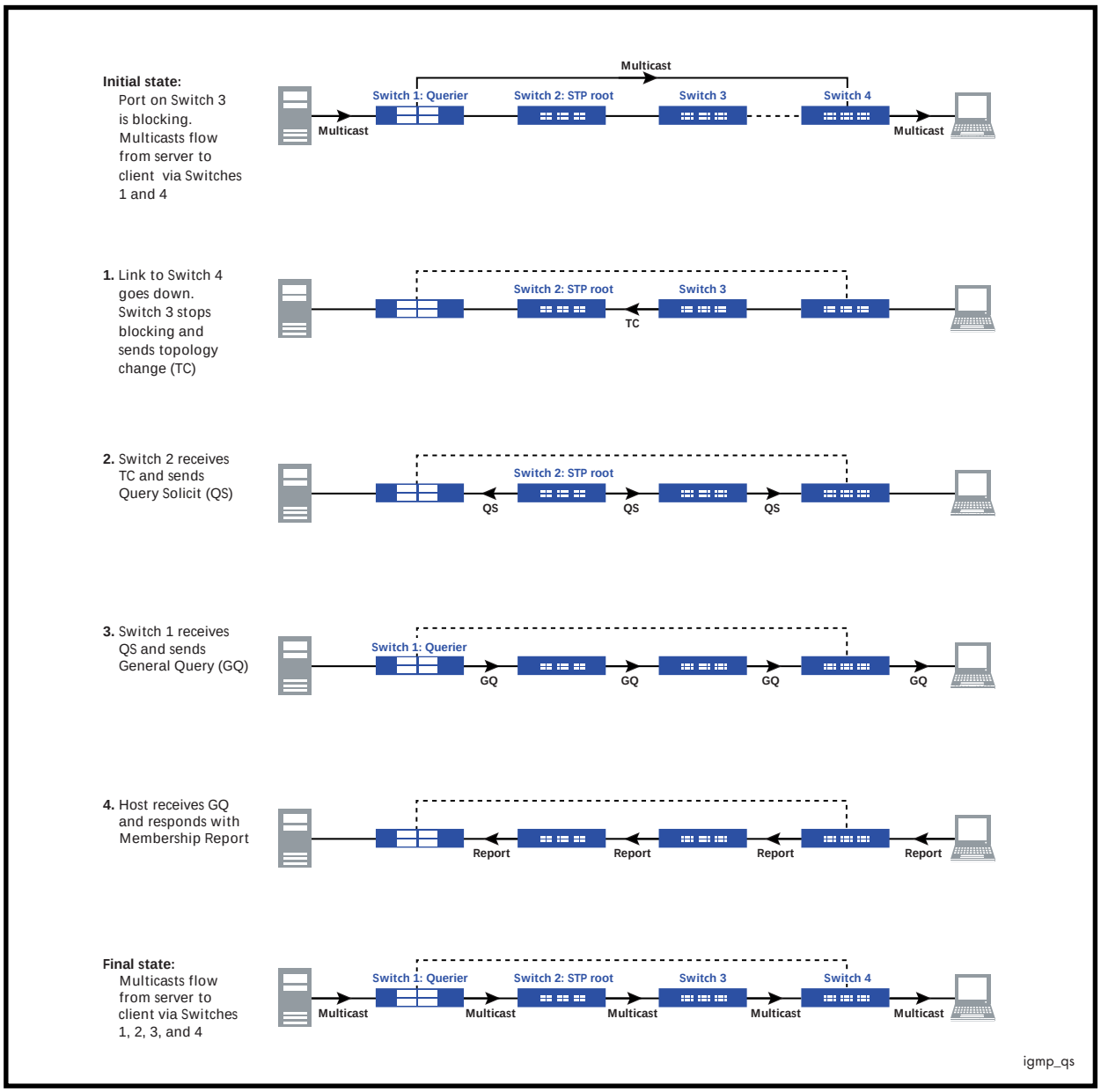
To get the network to converge faster, use the **ip igmp query-max-response-time** command and set a low response time value, such as one or two seconds, so that the clients will respond immediately with a report as a response to the IGMP Queries

On switches other than the STP root bridge or the EPSR master node, you can disable query solicitation by using the no variant of the **ip igmp snooping tcn query solicit** command. In addition, on all switches, you can disable query solicitation on a per-vlan basis using the no variant of the **ip igmp snooping tcn query solicit** command in Interface Configuration mode, after specifying a VLAN first in Interface Configuration mode.

To see whether query solicitation is on or off, check the Query Solicitation field in output of the **show ip igmp interface** command. You can view running and startup configurations with **show running-config** and **show startup-config** commands to see if Query Solicitation is enabled.

The following figure shows how Query Solicitation works when a port goes down.

Figure 48-3: Query Solicitation when a port goes down



Speeding up IGMP Convergence in a Non-Looped Topology

For loop-free networks running IGMP, where it may take up to two minutes for multicasting to recover in a non-looped topology after a port comes back up, you can speed up convergence by enabling RSTP using the **spanning-tree mode** and **spanning-tree enable** commands.

RSTP enables the network to use Query Solicitation by default, and means that multicasting should resume within seconds, not minutes, of the link coming up.

Enabling Query Solicitation on Multiple Switches in a Looped Topology

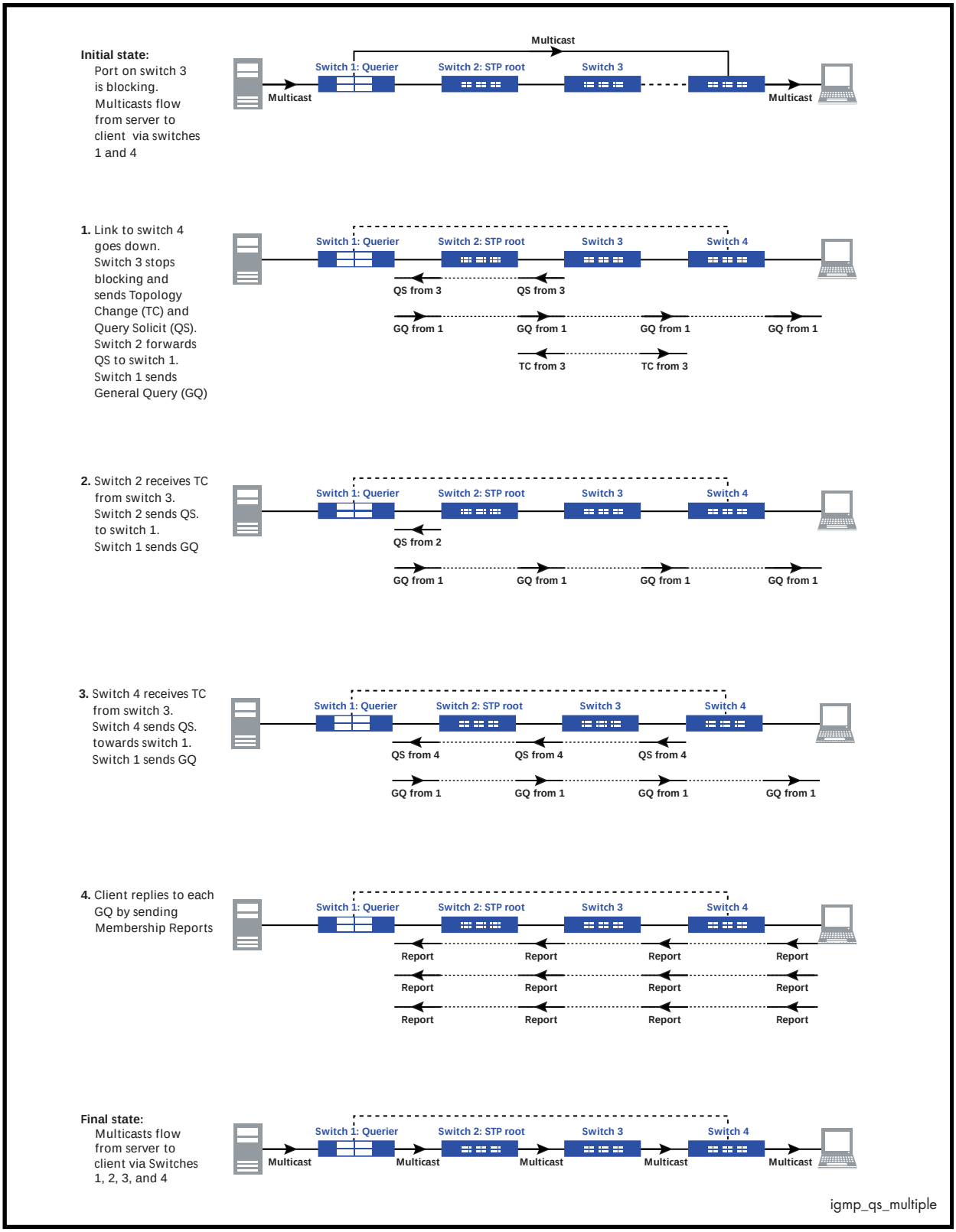
On networks that use spanning tree or EPSR, Query Solicitation is not normally required on switches other than the STP root bridge or EPSR master node. Therefore, it is only enabled by default on the root bridge and the master node.

However, in some networks you may need to turn on Query Solicitation on all switches - for example, if the network includes other switches that do not support Query Solicitation and therefore the STP root bridge may be a switch that does not send Query Solicit messages. To enable Query Solicitation, use the **ip igmp snooping tcn query solicit** command.

Every switch that has Query Solicitation enabled sends a Query Solicit message when it detects a topology change. Enabling it on multiple switches means you get multiple messages, but has no other disadvantage.

The following figure shows the packet flow for a four-switch network with Query Solicitation enabled on all the switches.

Figure 48-4: Packet flow for a four switch network with Query Solicitation enabled



Chapter 49: IGMP and IGMP Snooping Commands

Introduction 49.2
Command List 49.2
clear ip igmp 49.2
clear ip igmp group 49.3
clear ip igmp interface 49.4
debug igmp 49.5
ip igmp 49.6
ip igmp access-group 49.7
ip igmp immediate-leave 49.8
ip igmp last-member-query-count 49.9
ip igmp last-member-query-interval 49.10
ip igmp limit 49.11
ip igmp mroute-proxy 49.13
ip igmp proxy-service 49.14
ip igmp querier-timeout 49.15
ip igmp query-holdtime 49.16
ip igmp query-interval 49.18
ip igmp query-max-response-time 49.20
ip igmp ra-option (Router Alert) 49.22
ip igmp robustness-variable 49.23
ip igmp snooping 49.24
ip igmp snooping fast-leave 49.25
ip igmp snooping mrouter 49.26
ip igmp snooping querier 49.27
ip igmp snooping report-suppression 49.28
ip igmp snooping routermode 49.29
ip igmp snooping tcn query solicit 49.31
ip igmp source-address-check 49.33
ip igmp ssm 49.34
ip igmp ssm-map enable 49.35
ip igmp ssm-map static 49.36
ip igmp static-group 49.38
ip igmp startup-query-count 49.40
ip igmp startup-query-interval 49.41
ip igmp version 49.42
show debugging igmp 49.43
show ip igmp groups 49.44
show ip igmp interface 49.45
show ip igmp proxy 49.48
show ip igmp snooping mrouter 49.49
show ip igmp snooping routermode 49.50
show ip igmp snooping statistics 49.51
undebug igmp 49.51

Introduction

The Internet Group Management Protocol (IGMP) module includes the IGMP Proxy service and IGMP Snooping functionality. Some of the following commands may have commonalities and restrictions. These are described under the Usage section for each command.

Command List

This chapter provides an alphabetical reference of configure, clear, and show commands related to Internet Group Management Protocol (IGMP) and IGMP Snooping.

clear ip igmp

Use this command to clear all IGMP group membership records on all VLAN interfaces.

Syntax `clear ip igmp`

Mode Privileged Exec

Usage This command applies to VLAN interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
awplus# clear ip igmp
```

Validation Commands **show ip igmp interface**
show running-config

Related Commands **clear ip igmp group**
clear ip igmp interface

clear ip igmp group

Use this command to clear IGMP group membership records for a specific group on either all VLAN interfaces, a single VLAN interface, or for a range of VLAN interfaces.

Syntax

```
clear ip igmp group *
clear ip igmp group <ip-address> <interface>
```

Parameter	Description
*	Clears all groups on all VLAN interfaces. This is an alias to the **clear ip igmp** command.
<ip-address>	Specifies the group whose membership records will be cleared from all VLAN interfaces, entered in the form A.B.C.D.
<interface>	Specifies the name of the VLAN interface; all groups learned on this VLAN interface are deleted.

Mode

Privileged Exec

Usage

This command applies to groups learned by IGMP, IGMP Snooping, or IGMP Proxy.

In addition to the group a VLAN interface can be specified. Specifying this will mean that only entries with the group learned on the interface will be deleted.

Examples

```
awplus# clear ip igmp group *
awplus# clear ip igmp group 224.1.1.1 vlan1
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

clear ip igmp
clear ip igmp interface

clear ip igmp interface

Use this command to clear IGMP group membership records on a particular VLAN interface.

Syntax `clear ip igmp interface <interface>`

Parameter	Description
<interface>	Specifies the name of the VLAN interface. All groups learned on this VLAN interface are deleted.

Mode Privileged Exec

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
awplus# clear ip igmp interface vlan1
```

Validation Commands **show ip igmp interface**
show running-config

Related Commands **clear ip igmp**
clear ip igmp group

debug igmp

Use this command to enable debugging of either all IGMP or a specific component of IGMP.

Use the **no** variant of this command to disable all IGMP debugging, or debugging of a specific component of IGMP.

Syntax

```
debug igmp {all|decode|encode|events|fsm|tib}
no debug igmp {all|decode|encode|events|fsm|tib}
```

Parameter	Description
`all`	Enable or disable all debug options for IGMP
`decode`	Debug of IGMP packets that have been received
`encode`	Debug of IGMP packets that have been sent
`events`	Debug IGMP events
`fsm`	Debug IGMP Finite State Machine (FSM)
`tib`	Debug IGMP Tree Information Base (TIB)

Modes

Privileged Exec and Global Configuration

Usage

This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Example

```
awplus# configure terminal
awplus(config)# debug igmp all
```

Related Commands

show debugging igmp
undebug igmp

ip igmp

Use this command to enable IGMP on an interface. The command configures the device as an IGMP querier.

Use the **no** variant of this command to return all IGMP related configuration to the default on this interface.

Syntax

```
ip igmp
no ip igmp
```

Default Disabled

Mode Interface Configuration for a VLAN interface.

Usage This command can only be configured on VLAN interfaces, and will have no effect on IGMP Proxy or IGMP Snooping configuration.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Note An IP address must be assigned to the VLAN first, before this command will work.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp
```

Validation Commands **show ip igmp interface**
show running-config

ip igmp access-group

This command adds an access control list to a VLAN interface configured for IGMP, IGMP Snooping, or IGMP Proxy. The access control list is used to control and filter the multicast groups learned on the VLAN interface.

The **no** variant of this command disables the access control filtering on the interface.

Syntax

```
ip igmp access-group {<access-list-number>|<access-list-name>}
no ip igmp access-group
```

Parameter	Description
<access-list-number>	Standard IP access-list number, in the range <1-99>.
<access-list-name>	Standard IP access-list name.

Default By default there are no access lists configured on any interface.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to VLAN interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

This command applies to VLAN interfaces configured for IGMP or IGMP Snooping.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example In the following example, hosts serviced by VLAN interface `vlan2` can only join the group `225.2.2.2`:

```
awplus# configure terminal
awplus(config)# access-list 1 permit 225.2.2.2 0.0.0.0
awplus(config)# interface vlan2
awplus(config-if)# ip igmp access-group 1
```

ip igmp immediate-leave

In IGMP version 2, use this command to minimize the leave latency of IGMP memberships for specified multicast groups. The specified access list number or name defines the multicast groups in which the immediate leave feature is enabled.

Use the **no** variant of this command to disable this feature.

Syntax

```
ip igmp immediate-leave group-list {<access-list-number>|<access-
    list-number-expanded>|<access-list-name>}

no ip igmp immediate-leave
```

Parameter	Description
<access-list-number>	Access-list number, in the range <1-99>.
<access-list-number-expanded>	Access-list number (expanded range), in the range <1300-1999>.
<access-list-name>	Standard IP access-list name.

Default Disabled by default.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example The following example shows how to enable the immediate-leave feature on the VLAN interface `vlan2` for a specific range of multicast groups:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp immediate-leave group-list 34
awplus(config-if)# exit
awplus(config)# access-list 34 permit 225.192.20.0 0.0.0.255
```

Related Commands **ip igmp last-member-query-interval**

ip igmp last-member-query-count

Use this command to set the last-member query-count value for an interface.

Use the **no** variant of this command to return to the default on an interface.

Syntax

```
ip igmp last-member-query-count <2-7>
no ip igmp last-member-query-count
```

Parameter	Description
<2-7>	Last member query count value.

Default The default last member query count value is 2.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp last-member-query-count 3
```

Validation Commands **show ip igmp interface**
show running-config

Related Commands **ip igmp last-member-query-interval**
ip igmp startup-query-count

ip igmp last-member-query-interval

Use this command to configure the frequency at which the router sends IGMP group specific host query messages.

Use the **no** variant of this command to set this frequency to the default.

Syntax

```
ip igmp last-member-query-interval <interval>
no ip igmp last-member-query-interval
```

Parameter	Description
<interval>	The frequency in milliseconds, in the range <1000-25500>, at which IGMP group-specific host query messages are sent.

Default 1000 milliseconds

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example The following example changes the IGMP group-specific host query message interval to 2 seconds (`2000` milliseconds) for VLAN interface `vlan1`:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ip igmp last-member-query-interval 2000
```

Validation Commands **show ip igmp interface**
show running-config

Related Commands **ip igmp immediate-leave**
ip igmp last-member-query-count

ip igmp limit

Use this command to configure the limit on the maximum number of group membership entries for the device as a whole or for the specified interface (if in interface mode). Once the specified number of group memberships is reached, all further membership reports will be ignored. Optionally, you can configure an access-list to stop certain addresses from being subject to the limit.

The limit is dependent on the MTU (Maximum Transmission Unit) of the interface, which is the size in bytes of the largest packet that a network protocol can transmit. Typically for an Ethernet channel with an MTU of 1500 the igmp group membership limit will be 183 groups, because each igmp group membership is 8 bytes.

Use the **no** variant of this command to unset the limit and any specified exception access-list.

Syntax

```
ip igmp limit <limitvalue> [except {<access-list-number>|<access-
    list-number-expanded>|<access-list-name>}]

no ip igmp limit
```

Parameter	Description
<limitvalue>	<2-512> Maximum number of group membership entries.
<access-list-number>	Access-list number, in the range <1-99>.
<access-list-number-expanded>	Access-list number (expanded range), in the range <1300-1999>.
<access-list-name>	Standard IP access-list name.

Default The default limit, which is reset by the **no** variant of this command, is the same as maximum number of group membership entries that can be learned with the **ip igmp limit** command.

The default limit of group membership entries that can be learned is 512 entries.

Mode Global Configuration and Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP, IGMP Snooping, or IGMP Proxy.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples The following example configures an IGMP limit of `100` group membership entries across all interfaces on which IGMP is enabled, and excludes group `224.1.1.1` from this limitation:

```
awplus# configure terminal
awplus(config)# access-list 1 permit 224.1.1.1 0.0.0.0
awplus(config)# ip igmp limit 100 except 1
```

The following example configures an IGMP limit of 100 group membership entries on VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp limit 100
```

ip igmp mroute-proxy

Use this command to enable IGMP mroute proxy on this downstream interface and associate it with the upstream proxy service interface.

Use the **no** variant of this command to remove the association with the proxy-service interface.

Syntax

```
ip igmp mroute-proxy <interface>
no ip igmp mroute-proxy
```

Parameter	Description
<interface>	The name of the VLAN interface.

Mode Interface Configuration for a VLAN interface.

Usage You must also enable the IGMP proxy service on the upstream interface, using the **ip igmp proxy-service** command. You can associate one or more downstream mroute proxy interfaces on the device with a single upstream proxy service interface. This downstream mroute proxy interface listens for IGMP reports, and forwards them to the upstream IGMP proxy service interface.

IGMP Proxy does not work with other multicast routing protocols, such as PIM-SM or PIM-DM. This command applies to interfaces configured for IGMP Proxy.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example The following example configures the VLAN interface `vlan2` as the upstream proxy-service interface for the downstream `vlan3` interface.

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip igmp mroute-proxy vlan2
```

Related Commands **ip igmp proxy-service**

ip igmp proxy-service

Use this command to enable the VLAN interface to be the upstream IGMP proxy-service interface for the device. All associated downstream IGMP mroute proxy interfaces on this device will have their memberships consolidated on this proxy service interface, according to IGMP host-side functionality.

Use the **no** variant of this command to remove the designation of the VLAN interface as an upstream proxy-service interface.

Syntax

```
ip igmp proxy-service
no ip igmp proxy-service
```

Mode Interface Configuration for a VLAN interface.

Usage This command is used with the **ip igmp mroute-proxy** command to enable forwarding of IGMP reports to a proxy service interface for all forwarding entries for this interface. You must also enable the downstream IGMP mroute proxy interfaces on this device using the command **ip igmp mroute-proxy**.

IGMP Proxy does not work with other multicast routing protocols, such as PIM-SM or PIM-DM.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example The following example designates the VLAN interface `vlan1` as the upstream proxy-service interface.

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ip igmp proxy-service
```

Related Commands **ip igmp mroute-proxy**

ip igmp querier-timeout

Use this command to configure the timeout period before the device takes over as the querier for the VLAN interface after the previous querier has stopped querying.

Use the **no** variant of this command to restore the default.

Syntax

```
ip igmp querier-timeout <timeout>
no ip igmp querier-timeout
```

Parameter	Description
<timeout>	IGMP querier timeout interval value in seconds, in the range <1-65535>.

Default The default timeout interval is 255 seconds.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to VLAN interfaces configured for IGMP. The timeout value should not be less than the current active querier's general query interval.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example The following example configures the device to wait `130` seconds from the time it received the last query before it takes over as the querier for the VLAN interface `vlan20`:

```
awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# ip igmp querier-timeout 130
```

Validation Commands **show ip igmp interface**
show running-config

Related Commands **ip igmp query-interval**

ip igmp query-holdtime

This command sets the time that an IGMP Querier waits after receiving a query solicitation before it sends an IGMP Query. IGMP General Query messages will not be sent during the hold time interval.

Use the **no** variant of this command to return to the default query hold time period.

Syntax

```
ip igmp query-holdtime <interval>
no ip igmp query-holdtime
```

Parameter	Description
<interval>	Query interval value in milliseconds, in the range <100-5000>.

Default By default the delay before sending IGMP General Query messages is 500 milliseconds.

Mode Interface Configuration for a VLAN interface.

Usage Use this command to configure a value for the IGMP query hold time in the current network. IGMP Queries can be generated after receiving Query Solicitation (QS) packets and there is a possibility of a DoS (Denial of Service) attack if a stream of Query Solicitation (QS) packets are sent to the IGMP Querier, eliciting a rapid stream of IGMP Queries. This command applies to interfaces on which the switch is acting as an IGMP Querier.

Use the **ip igmp query-interval** command when a delay for IGMP general query messages is required and IGMP general query messages are required. The **ip igmp query-holdtime** command stops IGMP query messages during the configured holdtime interval, so the rate of IGMP Queries that can be sent out of an interface can be restricted.

See **"Query Solicitation" on page 48.9** for introductory information about the Query Solicitation feature.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Note

This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCS Stack.

Examples To set the IGMP query holdtime to `900` ms for `vlan20`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# ip igmp query-holdtime 900
```

To reset the IGMP query holdtime to the default (500 ms) for `vlan10`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# no ip igmp query-holdtime
```

Validation Commands
show ip igmp interface
show running-config

Related Commands
ip igmp query-interval
ip igmp snooping tcn query solicit

ip igmp query-interval

Use this command to configure the period for sending IGMP General Query messages.

The IGMP query interval specifies the time between IGMP General Query messages being sent.

Use the **no** variant of this command to return to the default query interval period.

Note The IGMP query interval must be greater than IGMP query maximum response time.

Syntax

```
ip igmp query-interval <interval>
no ip igmp query-interval
```

Parameter	Description
<interval>	Query interval value in seconds, in the range <2-18000>.

Default The default IGMP query interval is 125 seconds.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP. Note that the IGMP query interval is automatically set to a greater value than the IGMP query max response time.

For example, if you set the IGMP query max response time to 2 seconds using the **ip igmp query-max-response-time** command, and the IGMP query interval is currently less than 3 seconds, then the IGMP query interval period will be automatically reconfigured to be 3 seconds, so it is greater than the IGMP query maximum response time.

Use the **ip igmp query-interval** command when a non-default interval for IGMP General Query messages is required.

The **ip igmp query-holdtime** command can occasionally delay the sending of IGMP Queries.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples

The following example changes the period between IGMP host-query messages to 3 minutes (`180` seconds) for VLAN interface `vlan20`:

```
awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# ip igmp query-interval 180
```

The following example resets the period between sending IGMP host-query messages to the default (125 seconds) for VLAN interface `vlan20`:

```
awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# no ip igmp query-interval
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

ip igmp query-holdtime
ip igmp query-max-response-time
ip igmp startup-query-interval

ip igmp query-max-response-time

Use this command to configure the maximum response time advertised in IGMP Queries.

Use the **no** variant of this command to restore the default.

Note The IGMP query maximum response time must be less than the IGMP query interval.

Syntax

```
ip igmp query-max-response-time <response-time>
no ip igmp query-max-response-time
```

Parameter	Description
<response-time>	Response time value in seconds, in the range <1-3180>.

Default The default IGMP query maximum response time is 10 seconds.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP. Note that the IGMP query interval is automatically set to a greater value than the IGMP query maximum response time.

For example, if you set the IGMP query interval to 3 seconds using the **ip igmp query-interval** command, and the current IGMP query interval is less than 3 seconds, then the IGMP query maximum response time will be automatically reconfigured to be 2 seconds, so it is less than the IGMP query interval time.

To get the network to converge faster, use the **ip igmp query-max-response-time** command and set a low response time value, such as one or two seconds, so that the clients will respond immediately with a report as a response to the IGMP Queries.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples

The following example configures a maximum response time of `8` seconds for VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp query-max-response-time 8
```

The following example restores the default maximum response time of `10` seconds for VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip igmp query-max-response-time
```

Validation Commands

show ip igmp interface
show running-config

Related Commands

ip igmp query-interval

ip igmp ra-option (Router Alert)

Use this command to enable strict Router Alert (RA) option validation. With strict RA option enabled, IGMP packets without RA options are ignored.

Syntax

```
ip igmp ra-option
no ip igmp ra-option
```

Default The default state of RA validation is unset.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP and IGMP Snooping.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example

```
awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# ip igmp ra-option
```

ip igmp robustness-variable

Use this command to change the robustness variable value on a VLAN interface.

Use the **no** variant of this command to return to the default on an interface.

Syntax

```
ip igmp robustness-variable <1-7>
no ip igmp robustness-variable
```

Parameter	Description
<1-7>	The robustness variable value.

Default The default robustness variable value is 2.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for IGMP and IGMP Snooping.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples

```
awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# ip igmp robustness-variable 3

awplus# configure terminal
awplus(config)# interface vlan20
awplus(config-if)# no ip igmp robustness-variable 3
```

Validation Commands **show ip igmp interface**
show running-config

ip igmp snooping

Use this command to enable IGMP Snooping. When this command is used in the Global Configuration mode, IGMP Snooping is enabled at the switch level. When this command is used in Interface Configuration mode, IGMP Snooping is enabled for the specified VLANs.

Use the **no** variant of this command to either globally disable IGMP Snooping, or disable IGMP Snooping on a specified interface.

Note IGMP snooping cannot be disabled on an interface if IGMP snooping has already been disabled globally. IGMP snooping can be disabled on both an interface and globally if disabled on the interface first and then disabled globally.

Syntax

```
ip igmp snooping
no ip igmp snooping
```

Default By default, IGMP Snooping is enabled both globally and on all VLANs.

Mode Global Configuration and Interface Configuration for a VLAN interface.

Usage For IGMP snooping to operate on particular VLAN interfaces, it must be enabled both globally by using this command in Global Configuration mode, and on individual VLAN interfaces by using this command in Interface Configuration mode (both are enabled by default.)

Both IGMP snooping and MLD snooping must be enabled globally on the switch for IGMP snooping to operate. MLD snooping is also enabled by default. To enable it if it has been disabled, use the **ipv6 mld snooping** command on page 56.22 in Global Configuration mode.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples

```
awplus# configure terminal
awplus(config)# ip igmp snooping

awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp snooping
```

Related Commands

ipv6 mld snooping
show ip igmp interface
show running-config

ip igmp snooping fast-leave

Use this command to enable IGMP Snooping fast-leave processing. Fast-leave processing is analogous to immediate-leave processing. The IGMP group-membership entry is removed as soon as an IGMP leave group message is received, without sending out a group-specific query.

Use the **no** variant of this command to disable fast-leave processing.

Syntax

```
ip igmp snooping fast-leave
no ip igmp snooping fast-leave
```

Default IGMP Snooping fast-leave processing is disabled.

Mode Interface Configuration for a VLAN interface.

Usage This IGMP Snooping command can only be configured on VLAN interfaces.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example This example shows how to enable fast-leave processing on the VLAN interface vlan2:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp snooping fast-leave
```

Validation Commands **show ip igmp interface**
show running-config

ip igmp snooping mrouter

Use this command to statically configure the specified port as a multicast router port for IGMP Snooping for an interface. This command applies to interfaces configured for IGMP Snooping.

Use the **no** variant of this command to remove the static configuration of the port as a multicast router port.

Syntax

```
ip igmp snooping mrouter interface <port>
no ip igmp snooping mrouter interface <port>
```

Parameter	Description
<port>	The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).

Mode Interface Configuration for a VLAN interface.

Usage Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example This example shows the switch port interface `port1.0.2` statically configured to be a multicast router interface for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp snooping mrouter interface port1.0.2
```

Related Commands **show ip igmp snooping mrouter**

ip igmp snooping querier

Use this command to enable IGMP querier operation when no multicast routing protocol is configured. When enabled, the IGMP Snooping querier sends out periodic IGMP queries for all interfaces. This command applies to interfaces configured for IGMP Snooping.

Use the **no** variant of this command to disable IGMP querier configuration.

Syntax

```
ip igmp snooping querier
no ip igmp snooping querier
```

Mode Interface Configuration for a VLAN interface.

Usage The IGMP Snooping querier uses the `0.0.0.0` Source IP address because it only masquerades as a proxy IGMP querier for faster network convergence.

It does not start, or automatically cease, the IGMP Querier operation if it detects query message(s) from a multicast router.

If an IP address is assigned to a VLAN, which has IGMP querier enabled on it, then the IGMP Snooping querier uses the VLAN's IP address as the Source IP Address in IGMP queries.

The IGMP Snooping Querier will not stop sending IGMP Queries if there is another IGMP Snooping Querier in the network with a lower Source IP Address.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Note Do not enable the IGMP Snooping Querier feature on a Layer 2 switch when there is an operational IGMP Querier in the network.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp snooping querier
```

Validation Commands **show ip igmp interface**
show running-config

ip igmp snooping report-suppression

Use this command to enable report suppression for IGMP versions 1 and 2. This command applies to interfaces configured for IGMP Snooping.

Report suppression stops reports being sent to an upstream multicast router port when there are already downstream ports for this group on this interface.

Use the **no** variant of this command to disable report suppression.

Syntax

```
ip igmp snooping report-suppression
no ip igmp snooping report-suppression
```

Default Report suppression does not apply to IGMPv3, and is turned on by default for IGMPv1 and IGMPv2 reports.

Mode Interface Configuration for a VLAN interface.

Usage Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example This example shows how to enable report suppression for IGMPv2 reports for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp version 2
awplus(config-if)# ip igmp snooping report-suppression
```

Validation Commands **show ip igmp interface**
show running-config

ip igmp snooping routermode

Use this command to set the destination IP addresses as a router multicast address, according to the routermode (all multicast addresses, default multicast addresses, specified multicast addresses).

Use the **no** variant of this command to the default. You can also remove a specified IP address from a custom list of multicast addresses.

Syntax

```
ip igmp snooping routermode {all|default|ip|multicastrouter|address
   <ip-address>}
```

```
no ip igmp snooping routermode [address <ip-address>]
```

Parameter	Description
`all`	All reserved multicast addresses (224.0.0.x). Packets from all possible addresses in range 224.0.0.x are set as routers.
`default`	Default set of reserved multicast addresses. Packets from 224.0.0.1, 224.0.0.2, 224.0.0.4, 224.0.0.5, 224.0.0.6, 224.0.0.9, 224.0.0.13, 224.0.0.15 and 224.0.0.24 are set as routers.
`ip`	Custom reserved multicast addresses. Custom IP address in the 224.0.0.x range are set as router multicast addresses using the **ip igmp snooping routermode address** command.
`multicastrouter`	DVMRP (224.0.0.4) and PIM (224.0.0.13) multicast addresses are set as routers.
`address`	Specify the multicast address in the 224.0.0.x range for use after issuing an **ip igmp snooping routermode ip** command
`<ip-address>`	IPv4 multicast address (224.0.0.x)

Default The default routermode is **default** not **all** and shows the below reserved multicast addresses:

```
Router mode.............Def
Reserved multicast address
          224.0.0.1
          224.0.0.2
          224.0.0.4
          224.0.0.5
          224.0.0.6
          224.0.0.9
          224.0.0.13
          224.0.0.15
          224.0.0.24
```

Mode Global Configuration

Examples To set **ip igmp snooping routermode** for all default reserved addresses enter:

```
awplus(config)# ip igmp snooping routermode default
```

To remove the multicast address 224.0.0.5 from the custom list of multicast addresses enter:

```
awplus(config)# no ip igmp snooping routermode address
                224.0.0.5
```

Related commands **show ip igmp snooping routermode**

ip igmp snooping tcn query solicit

Use this command to enable IGMP (Internet Group Management Protocol) Snooping TCN (Topology Change Notification) Query Solicitation feature. When this command is used in the Global Configuration mode, Query Solicitation is enabled.

Use the **no** variant of this command to disable IGMP Snooping TCN Query Solicitation. When the no variant of this command is used in Interface Configuration mode, this overrides the Global Configuration mode setting and Query Solicitation is disabled.

Syntax

```
ip igmp snooping tcn query solicit

no ip igmp snooping tcn query solicit
```

Default IGMP Snooping TCN Query Solicitation is disabled by default on the switch, unless the switch is the Master Node in an EPSR ring, or is the Root Bridge in a Spanning Tree.

When the switch is the Master Node in an EPSR ring, or the switch is the Root Bridge in a Spanning Tree, then IGMP Snooping TCN Query Solicitation is enabled by default and cannot be disabled using the Global Configuration mode command. However, Query Solicitation can be disabled for specified VLANs using this command from the Interface Configuration mode. Select the VLAN you want to disable in Interface Configuration mode then issue the no variant of this command to disable the specified VLAN without disabling this feature for other VLANs.

Mode Global Configuration and Interface Configuration for a VLAN interface.

Usage Once enabled, if the switch is not an IGMP Querier, on detecting a topology change, the switch generates IGMP Query Solicit messages that are sent to all the ports of the vlan configured for IGMP Snooping on the switch.

On a switch that is not the Master Node in an EPSR ring or the Root Bridge in a Spanning Tree, Query Solicitation can be disabled using the **no** variant of this command after being enabled.

If the switch that detects a topology change is an IGMP Querier then the switch will generate an IGMP Query message.

Note that the **no** variant of this command when issued in Global Configuration mode has no effect on a switch that is the Master Node in an EPSR ring or on a switch that is a Root Bridge in a Spanning Tree. Query Solicitation is not disabled for the switch these instances. However, Query Solicitation can be disabled on a per-vlan basis from the Interface Configuration mode.

See the below state table that shows when Query Solicit messages are sent in these instances:

Command issued from Global Configuration	Switch is STP Root Bridge or the EPSR Master Node	Command issued from Interface Configuration	IGMP Query Solicit message sent on VLAN
No	Yes	Yes	Yes
Yes	Yes	No	No
Yes	Yes	Yes	Yes

See **"Query Solicitation" on page 48.9** for introductory information about the Query Solicitation feature.

To ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Note This command will function on your switch in the stand-alone mode. but is not supported when the switch forms part of a VCS Stack.

Examples This example shows how to enable IGMP Snooping TCN Query Solicitation on a switch:

```
awplus# configure terminal
awplus(config)# ip igmp snooping tcn query solicit
```

This example shows how to disable IGMP Snooping TCN Query Solicitation on a switch:

```
awplus# configure terminal
awplus(config)# no ip igmp snooping tcn query solicit
```

This example shows how to enable IGMP Snooping TCN Query Solicitation for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp snooping tcn query solicit
```

This example shows how to disable IGMP Snooping TCN Query Solicitation for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip igmp snooping tcn query solicit
```

Validation Commands **show ip igmp interface**
show running-config

Related Commands **ip igmp query-holdtime**

ip igmp source-address-check

This command enables the checking of the Source Address for an IGMP Report, rejecting any IGMP Reports originating on devices outside of the local subnet.

Use the **no** variant of this command to disable the checking of the Source Address for an IGMP Report, which allows IGMP Reports from devices outside of the local subnet.

Syntax

```
ip igmp source-address-check
no ip igmp source-address-check
```

Default Source address checking for IGMP Reports is enabled by default.

Mode Interface Configuration for a VLAN interface.

Usage This is a security feature, and should be enabled unless IGMP Reports from outside the local subnet are expected, for example, if Multicast VLAN Registration is active in the network.

The no variant of this command is required to disable the IGMP Report source address checking feature in networks that use Multicast VLAN Registration to allow IGMP Reports from devices outside of the local subnet.

To ensure your VLAN is configured first, see **Chapter 18, Configuring VLANs**.

Examples To deny IGMP Reports from outside the current subnet for the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp source-address-check
```

To allow IGMP Reports from outside the current subnet for the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip igmp source-address-check
```

Validation Commands **show ip igmp interface**
show running-config

ip igmp ssm

Use this command to define a non-default Source Specific Multicast (SSM) range of IP multicast addresses in IGMP. Incoming IGMPv1 and IGMPv2 join requests are ignored if the multicast IP address is in the SSM range and no SSM mapping is configured for these addresses. By default, the SSM range is 232/8. To define the SSM range to be other than the default, use one of the access-list parameter options.

Use the **no** variant of this command to change the SSM range in IGMP back to the default.

Syntax

```
ip igmp ssm range {<access-list-number>|<access-list-name>}
no ip igmp ssm
```

Parameter	Description
<access-list-number>	Access-list number, in the range <1-99>.
<access-list-name>	Standard IP access-list name.

Default By default the SSM range is 232/8.

Mode Global Configuration

Examples To configure a non-default SSM range to be used in IGMP enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 10 permit 224.1.1.0 0.0.0.255
awplus(config)# ip igmp ssm range 10
```

To return to the default configuration enter the commands:

```
awplus# configure terminal
awplus(config)# no ip igmp ssm
```

Related Commands **access-list (standard numbered)**
ip pim ssm

ip igmp ssm-map enable

Use this command to enable Source Specific Multicast (SSM) mapping on the device.

Use the **no** variant of this command to disable SSM mapping.

Syntax

```
ip igmp ssm-map enable
no ip igmp ssm-map enable
```

Mode Global Configuration

Usage This command applies to VLAN interfaces configured for IGMP.

Example To enable SSM on the switch enter the commands:

```
awplus# configure terminal
awplus(config)# ip igmp ssm-map enable
```

Related Commands ip igmp ssm-map static

ip igmp ssm-map static

Use this command to specify the static mode of defining Source Specific Multicast (SSM) mapping. SSM statically assigns sources to IGMPv1 and IGMPv2 groups to translate such (*,G) groups' memberships to (S,G) memberships for use with PIM-SSM.

Use the **no** variant of this command to remove the SSM map association.

Syntax

```
ip igmp ssm-map static {<access-list-number>|<access-list-number-expanded>|<access-list-name>} <ip-address>
no ip igmp ssm-map static {<access-list-number>|<access-list-number-expanded>|<access-list-name>} <ip-address>
```

Parameter	Description
<access-list-number>	Access-list number, in the range <1-99>.
<access-list-number-expanded>	Access-list number (expanded range), in the range <1300-1999>.
<access-list-name>	Standard IP access-list name.
<ip-address>	Source address to use for static map group, entered in the form A.B.C.D.

Mode Global Configuration

Usage This command applies to VLAN interfaces configured for IGMP. You can use Standard numbered and Standard named ACLs plus Expanded Numbered ACLs.

Examples This example shows how to configure an SSM static mapping for group-address `224.1.1.1`, using a standard numbered ACL shown as `10`:

```
awplus# configure terminal
awplus(config)# access-list 10 permit 224.1.1.1 0.0.0.0
awplus(config)# ip igmp ssm-map static 10 1.2.3.4
```

This example shows how to configure an SSM static mapping for group-address `224.1.1.1`, using an expanded numbered ACL shown as `1301`:

```
awplus# configure terminal
awplus(config)# access-list 1301 permit 224.1.1.1 0.0.0.0
awplus(config)# ip igmp ssm-map static 1301 1.2.3.4
```

This example shows how to configure an SSM static mapping for group-address `224.1.1.1`, using a standard named ACL shown as `sales`:

```
awplus# configure terminal
awplus(config)# access-list sales permit 224.1.1.1 0.0.0.0
awplus(config)# ip igmp ssm-map static sales 1.2.3.4
```

Related Commands **ip igmp ssm-map enable**

ip igmp static-group

Use this command to statically configure multicast group membership entries on a VLAN interface, or to statically forward a multicast channel out a particular port or port range.

To statically add only a group membership, do not specify any parameters.

To statically add a (*,g) entry to forward a channel out of a port, specify only the multicast group address and the switch port range.

To statically add an (s,g) entry to forward a channel out of a port, specify the multicast group address, the source IP address, and the switch port range.

To use Source Specific Multicast mapping to determine the source IP address of the multicast server use the **ssm-map** parameter instead of specifying the source IP address.

Use the **no** variant of this command to delete static group membership entries.

Syntax

```
ip igmp static-group <ip-address> [source {<ip-source-addr>|ssm-map}]
    [interface <port>]

no ip igmp static-group <ip-address> [source {<ip-source-addr>|
    ssm-map}] [interface <port>]
```

Parameter	Description
<ip-address>	Standard IP Multicast group address, entered in the form A.B.C.D, to be configured as a static group member.
`source`	Optional.
<ip-source-addr>	Standard IP source address, entered in the form A.B.C.D, to be configured as a static source from where multicast packets originate.
`ssm-map`	This parameter uses Source Specific Multicast (SSM) Mapping to determine the source IP address associated with the specified IP Multicast group address. SSM mappings are configured using the **ip igmp ssm-map static** command.
`interface`	Use this parameter to specify a specific switch port or switch port range to statically forward the multicast group out of. If not used, static configuration is applied on all ports in the VLAN.
<port>	The port or port range to statically forward the group out of. The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa2`), or a dynamic (LACP) channel group (e.g. `po2`).

Mode Interface Configuration for a VLAN interface.

Usage This command applies to IGMP operation on a specific interface to statically add group and/or source records, or to IGMP Snooping on a VLAN interface to statically add group and/or source records.

To ensure your VLAN is configured first, see **Chapter 18, Configuring VLANs**.

Example The following example show how to statically add group and source records for IGMP on the VLAN interface `vlan3`:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip igmp
awplus(config-if)# ip igmp static-group 226.1.2.4 source
10.2.3.4
```

ip igmp startup-query-count

Use this command to configure the IGMP startup query count for an interface. The IGMP startup query count is the number of IGMP General Query messages sent by a querier at startup. The default IGMP startup query count is 2.

Use the **no** variant of this command to return an interface's configured IGMP startup query count to the default.

Syntax

```
ip igmp startup-query-count <startup-query-count>
no ip igmp startup-query-count
```

Parameter	Description
<startup-query-count>	Specify the IGMP startup query count for a VLAN interface in the range <2-10> where 2 is the default IGMP query count.

Default The default IGMP startup query count is 2.

Mode Interface Configuration for a VLAN interface.

Usage Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples The following example shows how to configure the IGMP startup query count to 4 for the VLAN interface vlan3:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# ip igmp startup-query-count 4
```

The following example shows how to remove the IGMP startup query count for the VLAN interface vlan3:

```
awplus# configure terminal
awplus(config)# interface vlan3
awplus(config-if)# no ip igmp startup-query-count
```

Related Commands
ip igmp last-member-query-count
ip igmp startup-query-interval

ip igmp startup-query-interval

Use this command to configure the IGMP startup query interval for an interface. The IGMP startup query interval is the amount of time in seconds between successive IGMP General Query messages sent by a querier during startup. The default IGMP startup query interval is one quarter of the IGMP query interval value.

Use the **no** variant of this command to return an interface's configured IGMP startup query interval to the default.

Syntax

```
ip igmp startup-query-interval <startup-query-interval>
no ip igmp startup-query-interval
```

Parameter	Description
<startup-query-interval>	Specify the IGMP startup query interval for a VLAN interface in Interface Configuration mode in the range of <2-1800> seconds to be one quarter of the IGMP query interval value.

Default The default IGMP startup query interval is one quarter of the IGMP query interval value.

Note The IGMP startup query interval must be one quarter of the IGMP query interval.

Mode Interface Configuration for a VLAN interface.

Usage Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Examples The following example shows how to configure the IGMP startup query interval to `15` seconds for the VLAN interface `vlan2` to be one quarter of the IGMP query interval value of 60 seconds:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip igmp startup-query-interval 15
awplus(config-if)# ip igmp query-interval 60
```

The following example shows how to remove the IGMP startup query interval for the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip igmp startup-query-interval
```

Related Commands
ip igmp last-member-query-interval
ip igmp query-interval
ip igmp startup-query-count

ip igmp version

Use this command to set the current IGMP version (IGMP version 1, 2 or 3) on an interface.

Use the **no** variant of this command to return to the default version.

Syntax

```
ip igmp version <1-3>
no ip igmp version
```

Parameter	Description
<1-3>	IGMP protocol version number

Default The default IGMP protocol version number is 3.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to VLAN interfaces configured for IGMP.

Ensure your VLAN is configured first: see **Chapter 18, Configuring VLANs**.

Example

```
awplus# configure terminal
awplus(config)# interface vlan5
awplus(config-if)# ip igmp version 2
```

Validation Commands **show ip igmp interface**

show debugging igmp

Use this command to display the IGMP debugging options set.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging igmp`

Mode User Exec and Privileged Exec

Example To display the IGMP debugging options set, enter the command:

```
awplus# show debugging igmp
```

Output **Figure 49-1: Example output from the show debugging igmp command**

```
IGMP Debugging status:
  IGMP Decoder debugging is on
  IGMP Encoder debugging is on
  IGMP Events debugging is on
  IGMP FSM debugging is on
  IGMP Tree-Info-Base (TIB) debugging is on
```

Related Commands **debug igmp**

show ip igmp groups

Use this command to display the multicast groups with receivers directly connected to the router, and learned through IGMP.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip igmp groups [<ip-address>|<interface> detail]`

Parameter	Description
<ip-address>	Address of the multicast group, entered in the form A.B.C.D.
<interface>	Interface name for which to display local information.

Mode User Exec and Privileged Exec

Example The following command displays local-membership information for all ports in all interfaces:

```
awplus# show ip igmp groups
```

Output **Figure 49-2: Example output from the show ip igmp groups command**

```
IGMP Connected Group Membership
Group Address    Interface       Uptime    Expires   Last
Reporter
224.0.1.1        port1.0.1       00:00:09  00:04:17  10.10.0.82
224.0.1.24       port1.0.2       00:00:06  00:04:14  10.10.0.84
224.0.1.40       port1.0.3       00:00:09  00:04:15  10.10.0.91
224.0.1.60       port1.0.3       00:00:05  00:04:15  10.10.0.7
224.100.100.100  port1.0.1       00:00:11  00:04:13  10.10.0.91
228.5.16.8       port1.0.3       00:00:11  00:04:16  10.10.0.91
228.81.16.8      port1.0.7       00:00:05  00:04:15  10.10.0.91
228.249.13.8     port1.0.3       00:00:08  00:04:17  10.10.0.91
235.80.68.83     port1.0.11      00:00:12  00:04:15  10.10.0.40
239.255.255.250  port1.0.3       00:00:12  00:04:15  10.10.0.228
239.255.255.254  port1.0.12      00:00:08  00:04:13  10.10.0.84
```

Table 49-1: Parameters in the output of the show ip igmp groups command

Parameter	Description
`Group Address`	Address of the multicast group.
`Interface`	Port through which the group is reachable.
`Uptime`	The time in weeks, days, hours, minutes, and seconds that this multicast group has been known to the device.
`Expires`	Time (in hours, minutes, and seconds) until the entry expires.
`Last Reporter`	Last host to report being a member of the multicast group.

show ip igmp interface

Use this command to display the state of IGMP, IGMP Proxy service, and IGMP Snooping for a specified VLAN, or all VLANs. IGMP is shown as Active or Disabled in the show output.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip igmp interface [<interface>]`

Parameter	Description
<interface>	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Examples The following output shows IGMP interface status for **vlan2** (with IGMP Snooping enabled):

```
awplus#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
awplus(config)#interface vlan2
awplus(config-if)#ip igmp snooping
awplus(config-if)#exit
awplus(config)#exit
awplus#show ip igmp interface vlan2
Interface vlan2 (Index 202)
 IGMP Disabled, Inactive, Version 3 (default)
 IGMP interface has 0 group-record states
 IGMP activity: 0 joins, 0 leaves
 IGMP robustness variable is 2
 IGMP last member query count is 2
 IGMP query interval is 125 seconds
 IGMP query holdtime is 500 milliseconds
 IGMP querier timeout is 255 seconds
 IGMP max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
 Strict IGMPv3 ToS checking is disabled on this interface
 Source Address checking is enabled
 IGMP Snooping is globally enabled
 IGMP Snooping query solicitation is globally disabled
  Num. query-solicit packets: 57 sent, 0 recvd
 IGMP Snooping is enabled on this interface
 IGMP Snooping fast-leave is not enabled
 IGMP Snooping querier is not enabled
 IGMP Snooping report suppression is enabled
awplus#
```

The following output shows IGMP interface status for **vlan2** (with IGMP Snooping disabled):

```
awplus#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
awplus(config)#interface vlan2
awplus(config-if)#no ip igmp snooping
awplus(config-if)#exit
awplus(config)#exit
awplus#show ip igmp interface vlan2
Interface vlan2 (Index 202)
 IGMP Disabled, Inactive, Version 3 (default)
 IGMP interface has 0 group-record states
 IGMP activity: 0 joins, 0 leaves
 IGMP robustness variable is 2
 IGMP last member query count is 2
 IGMP query interval is 125 seconds
 IGMP query holdtime is 500 milliseconds
 IGMP querier timeout is 255 seconds
 IGMP max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
 Strict IGMPv3 ToS checking is disabled on this interface
 Source Address checking is enabled
 IGMP Snooping is globally enabled
 IGMP Snooping query solicitation is globally disabled
  Num. query-solicit packets: 57 sent, 0 recvd
 IGMP Snooping is not enabled on this interface
 IGMP Snooping fast-leave is not enabled
 IGMP Snooping querier is not enabled
 IGMP Snooping report suppression is enabled
awplus#
```

The following command displays the IGMP interface status and Query Solicitation for **vlan3**:

```
awplus#show ip igmp interface vlan3
Interface vlan3 (Index 203)
 IGMP Enabled, Active, Querier, Version 3 (default)
 Internet address is 192.168.9.1
 IGMP interface has 256 group-record states
 IGMP activity: 51840 joins, 0 leaves
 IGMP robustness variable is 2
 IGMP last member query count is 2
 IGMP query interval is 125 seconds
 IGMP query holdtime is 500 milliseconds
 IGMP querier timeout is 250 seconds
 IGMP max query response time is 1 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 251 seconds
 Strict IGMPv3 ToS checking is disabled on this interface
 IGMP Snooping is globally enabled
 IGMP Snooping query solicitation is globally enabled
  Num. query-solicit packets: 1 sent, 10 recvd
 IGMP Snooping is enabled on this interface
 IGMP Snooping fast-leave is not enabled
 IGMP Snooping querier is not enabled
 IGMP Snooping report suppression is enabled
awplus#
```

Note

Query Solicitation status information is highlighted in **bold** in the above output.

Use the **show ip igmp interface** command to validate that Query Solicitation is enabled and to show the number of query-solicit message packets sent and received on a VLAN.

Related Commands

clear ip igmp
clear ip igmp group
clear ip igmp interface
ip igmp
ip igmp last-member-query-count
ip igmp last-member-query-interval
ip igmp querier-timeout
ip igmp query-holdtime
ip igmp query-interval
ip igmp query-max-response-time
ip igmp robustness-variable
ip igmp snooping
ip igmp snooping fast-leave
ip igmp snooping querier
ip igmp snooping report-suppression
ip igmp snooping tcn query solicit
ip igmp version

show ip igmp proxy

Use this command to display the state of IGMP Proxy services for a specified interface or for all interfaces.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip igmp proxy
show ip igmp proxy groups [detail]
show ip igmp proxy groups <multicast-group> [detail]
show ip igmp proxy groups <vlan> [detail]
show ip igmp proxy groups <vlan> <multicast-group> [detail]
```

Parameter	Description
`groups`	Specify IGMP proxy group membership information.
`detail`	Specify detailed IGMPv3 source information.
<vlan>	Specify the name of a single VLAN interface, for example **vlan1**.
<multicast-group>	Specify the IPv4 address in of the multicast group, in the format A.B.C.D.

Mode User Exec and Privileged Exec

Example To display the state of IGMP Proxy services for all interfaces, enter the command:

```
awplus# show ip igmp proxy
```

To display the state of IGMP Proxy services for VLAN interface **vlan1**, enter the command:

```
awplus# show ip igmp proxy groups vlan1
```

To display the detailed state of IGMP Proxy services for VLAN interface **vlan1**, enter the command:

```
awplus# show ip igmp proxy groups vlan1 detail
```

Related Commands **ip igmp proxy-service**

show ip igmp snooping mrouter

Use this command to display the multicast router ports, both static and dynamic, in a VLAN.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip igmp snooping mrouter [interface <interface>]`

Parameter	Description
`interface`	A specific interface.
`<interface>`	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Example To show all multicast router interfaces, use the command:

```
awplus# show ip igmp snooping mrouter
```

To show the multicast router interfaces in `vlan1`, use the command:

```
awplus# show ip igmp snooping mrouter interface vlan1
```

Output **Figure 49-3: Example output from the show ip igmp snooping mrouter command**

```
VLAN    Interface   Static/Dynamic
1       port1.0.5   Statically configured
200     port1.0.2   Statically configured
```

Figure 49-4: Example output from the show ip igmp snooping mrouter interface vlan1 command

```
VLAN    Interface   Static/Dynamic
1       port1.0.5   Statically configured
```

Related Commands **ip igmp snooping mrouter**

show ip igmp snooping routermode

Use this command to display the current routermode and the list of IP addresses set as router multicast addresses from the **ip igmp snooping routermode** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip igmp snooping routermode`

Mode User Exec and Privileged Exec

Example To show the routermode and the list of router multicast addresses, use the command:

```
awplus# show ip igmp snooping routermode
```

Output **Figure 49-5: Example output from the show ip igmp snooping routermode command**

```
Router mode.............Def
Reserved multicast address
         224.0.0.1
         224.0.0.2
         224.0.0.4
         224.0.0.5
         224.0.0.6
         224.0.0.9
         224.0.0.13
         224.0.0.15
         224.0.0.24
```

Related Commands **ip igmp snooping routermode**

show ip igmp snooping statistics

Use this command to display IGMP Snooping statistics data.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip igmp snooping statistics interface <interface-range>
   [group [<ip-address>]]
```

Parameter	Description
<ip-address>	Optionally specify the address of the multicast group, entered in the form A.B.C.D.
<interface>	Specify the name of the VLAN interface or interface range.

Mode User Exec and Privileged Exec

Example To display IGMP statistical information for **vlan1** and **vlan2**, use the command:

```
awplus# show ip igmp snooping statistics interface
        vlan1-vlan2
```

Output **Figure 49-6: Example output from the show ip igmp snooping statistics command**

```
IGMP Snooping statistics for vlan1
Interface:      port1.0.3
Group:          224.1.1.1
Uptime:         00:00:09
Group mode:     Exclude (Expires: 00:04:10)
Last reporter:  10.4.4.5
Source list is empty
IGMP Snooping statistics for vlan2
Interface:      port1.0.4
Group:          224.1.1.2
Uptime:         00:00:19
Group mode:     Exclude (Expires: 00:05:10)
Last reporter:  10.4.4.6
Source list is empty
```

undebug igmp

This command applies the functionality of the no **debug igmp** command on page 49.5.

Chapter 50: PIM-SM Introduction and Configuration

Introduction 50.2
PIM-SM 50.2
Characteristics of PIM-SM 50.2
Roles in PIM-SM 50.3
Operation of PIM-SM 50.4
PIM-SM Configuration 50.6
Static Rendezvous Point configuration 50.7
Dynamic Rendezvous Point Configuration 50.9
Bootstrap Router Configuration 50.10
PIM-SSM 50.13
Characteristics of PIM-SSM 50.13
PIM-SSM IP Address Range 50.13
IGMPv3 and SSM-Mapping 50.13
How PIM-SSM Works 50.14
How IGMP SSM-Mapping Works 50.15
Configure PIM-SSM 50.15

Introduction

This chapter provides information about Protocol Independent Multicast - Sparse Mode (PIM-SM) and Protocol Independent Multicast - Source Specific Multicast (PIM-SSM).

PIM-SM

Protocol Independent Multicast - Sparse Mode (PIM-SM) provides efficient communication between members of sparsely distributed groups - the type of groups that are most common in wide-area internetworks.

For details of the commands used to configure PIM-SM, see **Chapter 51, PIM-SM Commands**. For a general overview of multicasting, see **Chapter 47, Multicast Introduction and Commands**.

Characteristics of PIM-SM

PIM Sparse Mode (PIM-SM) is designed on the principle that several hosts wishing to participate in a multicast conference does not justify flooding the entire internetwork with periodic multicast traffic. PIM-SM is designed to limit multicast traffic so that only those switches interested in receiving traffic for a particular group receive the traffic.

Switches with directly attached or downstream members are required to join a Sparse Mode distribution tree by transmitting explicit join messages. If a switch does not become part of the predefined distribution tree, it does not receive multicast traffic addressed to the group. In contrast, dense mode multicast routing protocols assume downstream group membership and continue to forward multicast traffic on downstream links until explicit prune messages are received. The default forwarding action of a sparse mode multicast routing protocol is to block traffic unless it is explicitly requested, while the default action of the dense mode multicast routing protocols is to forward traffic.

PIM-SM employs the concept of a Rendezvous Point (RP) where receivers “meet” sources. The initiator of each multicast group selects a primary RP and a small ordered set of alternative RPs, known as the RP-list. For each multicast group, there is only a single active RP. Each receiver wishing to join a multicast group contacts its directly attached switch, which in turn joins the multicast distribution tree by sending an explicit join message to the group’s primary RP. A source uses the RP to announce its presence and to find a path to members that have joined the group. This model requires Sparse Mode switches to maintain some state information (the RP-list) prior to the arrival of data packets. In contrast, Dense Mode multicast routing protocols are data driven, since they do not define any state for a multicast group until the first data packet arrives.

Roles in PIM-SM

A multicast sender does not need to know the addresses of the members of the group in order to send to them, and the members of the group need not know the address of the sender. Group membership can change at any time. When PIM is enabled on the switch, and before the switch can route multicast traffic, it must establish which of the PIM routers in the network are performing some key roles:

- Designated Router.
- Rendezvous Point.
- Bootstrap Router.

Designated Router

There must be one PIM Designated Router (DR) in the subnetwork to which the IP hosts are connected. Any PIM-SM interfaces on the subnetwork elect the DR with the highest DR priority. If there is more than one router with the same priority, or no priority, they choose the interface with the highest IP address number. The DR performs all the PIM functionality for the subnetwork. If the current DR becomes unavailable, the remaining switches elect a new DR on the interface by DR priority or IP address.

Rendezvous Point

Each multicast group must have a Rendezvous Point (RP). The RP forms the root of the group's distribution tree. The DR for a multicast sender sends multicast packets towards the RP. DRs with group members connected to them send join messages towards the group's RP. The RP candidate with the lowest priority is elected from all the RP candidates for a group. If the RP becomes unavailable, the remaining RP candidates elect a new RP.

Bootstrap Router

Each PIM-SM network must have at least one Bootstrap Router (BSR) candidate, unless all switches in the domain are configured statically with information about all RPs in the domain. Every switch that is a BSR candidate periodically sends a Bootstrap Candidate Advertisement message to advertise that it is available as a BSR candidate. The BSR candidates in the network elect the switches with the highest preference value to be the BSR. The elected BSR listens to PIM Candidate RP Advertisement messages specifying RP candidates for multicast groups. It maintains a list of RP candidates and sends a bootstrap message every BSM interval, specifying all the multicast groups in the PIM network, and their RP candidates. Each switch uses this information and a standardized hash mechanism to determine the RP for each group.

In summary:

- Each multicast group must have at least one RP candidate
- Each PIM-SM domain must have at least one BSR candidate, unless all routers in the domain are configured statically with information about all RPs in the domain
- Each subnetwork must have at least one DR candidate.

PIM hello messages

When PIM is enabled on a switch, it sends out a PIM Hello message on all its PIM enabled interfaces, and listens for Hello messages from its PIM neighbors. When a switch receives a Hello message, it records the interface, IP address, priority for becoming a DR, and the timeout for the neighbor's information. The switch sends Hello messages regularly at the Hello Time interval.

Operation of PIM-SM

Once roles are established, multicast routing follows specific phases:

1. **Rendezvous Point Tree**
2. **Register stop**
3. **Shortest Path Tree**

While multicast routing always begins with phase 1, the Designated Router (DR) for a receiver determines whether and when to move on to phases 2 and 3, depending on the amount of traffic from the source.

Rendezvous Point Tree

Phase 1 establishes and uses a shared tree rooted at the Rendezvous Point (RP) to forward all multicast data to group members.

When an IP host sends an IGMP join message to the local PIM DR, which is not the RP for the group, the DR sends a PIM join message towards the RP for the group ("upstream"). The DR determines which switch is the RP for the group from the most recent bootstrap message. Every switch the join message passes through records that there is a group member on the incoming interface. Eventually, the join message reaches either the RP, or another switch that already knows that it has a group member downstream. If the group has many members, the join messages converge on the RP to form a Rendezvous Point Tree (RPT). This is called a shared tree because multicast data that is sent to the group by any sender shares the tree. The multicast receiver's DR sends join messages periodically according to the upstream join timer as long as the IP host is a member of the group. When the last receiver on a subnetwork leaves the group, the join messages stop, and their entries timeout on routers that are closer to the RP.

The sender's DR encapsulates the multicast data in a unicast packet in a process called **registering**, and sends these register packets to the group's RP. When the RP receives the data, it decapsulates them, and forwards them onto the shared tree.

Register stop

Phase 2 improves efficiency and performance by using register stop. In this phase the RP joins the shortest path tree between the source and receiver. This allows the original (unencapsulated) packets to be forwarded from the sender, instead of encapsulated packets. It also allows shorter paths to receivers that are close to the sender, making it more efficient in some circumstances.

When the RP for a group receives the first encapsulated data packet from a source, it joins the shortest path tree towards the sender. Once data is able to flow along the shortest path from the sender to the RP, packets do not need to be registered. The RP sends a register stop message in reply to the next encapsulated message. When the sender's DR receives the register stop message, it stops registering. The DR sends a null register message to the RP to find whether the RP still does not need to receive registered packets. If it receives another register stop message, the DR continues to forward only the native data packets. If the DR does not receive another register stop message within the register probe time, it resumes registering the data packets and sending them to the RP.

When the RP starts receiving native data packets from the source, it starts to discard the encapsulated packets, and starts forwarding native packets on the shared tree to all the group members. If the path from the source to the RP intersects the shared RP tree for the group, then the packets also take a short-cut onto the shared tree for delivery to the group members down its branches.

Shortest Path Tree

This phase further optimizes routing by using Shortest Path Trees (SPT). In phase 3 the receiver joins the shortest path tree between the source and receiver. This allows a multicast group member to receive multicast data by the shortest path from the sender, instead of from the shared RP tree. When the receiver's DR receives multicast data from a particular sender, it sends a join message towards the sender. When this message reaches the sender's DR, the DR starts forwarding the multicast data directly towards the receiver. As several receivers all initiate shortest paths to the sender, these paths converge, creating a SPT.

When the multicast packets start arriving from the SPT at the receiver's DR or an upstream router common to the SPT and the RPT, it starts discarding the packets from the RPT, and sends a prune message towards the RP. The prune message travels up the RPT until it reaches the RP or a switch that still needs to forward multicast packets from this sender to other receivers. Every time a switch receives a prune message, it waits a short time so that other switches on the LAN have the opportunity to override the prune message.

Multi-Access LANs

If the PIM-SM network includes multi-access LAN links for transit, as well as point-to-point links, then a mechanism is needed to prevent multiple trees forwarding the same data to the same group member. Two or more switches on a LAN may have different information about how to reach the RP or the multicast sender. They could each send a join message to two different switches closer to the RP for an RPT or the sender for an SPT. This could potentially cause two copies of all the multicast traffic towards the receiver.

When PIM switches notice duplicate data packets on the LAN, they elect a single switch to forward the data packets, by each sending PIM Assert messages. If one of the upstream switches is on an SPT and the other is on an RPT, the switch on the SPT has the shortest path to the sender, and wins the Assert election. If both switches are on RPTs the switch with the shortest path to the RP (the lowest sum of metrics to the RP) wins the Assert. If both switches are on an SPT, then the switch with the shortest path to the sender (the lowest sum of metrics to the sender's DR) wins the Assert.

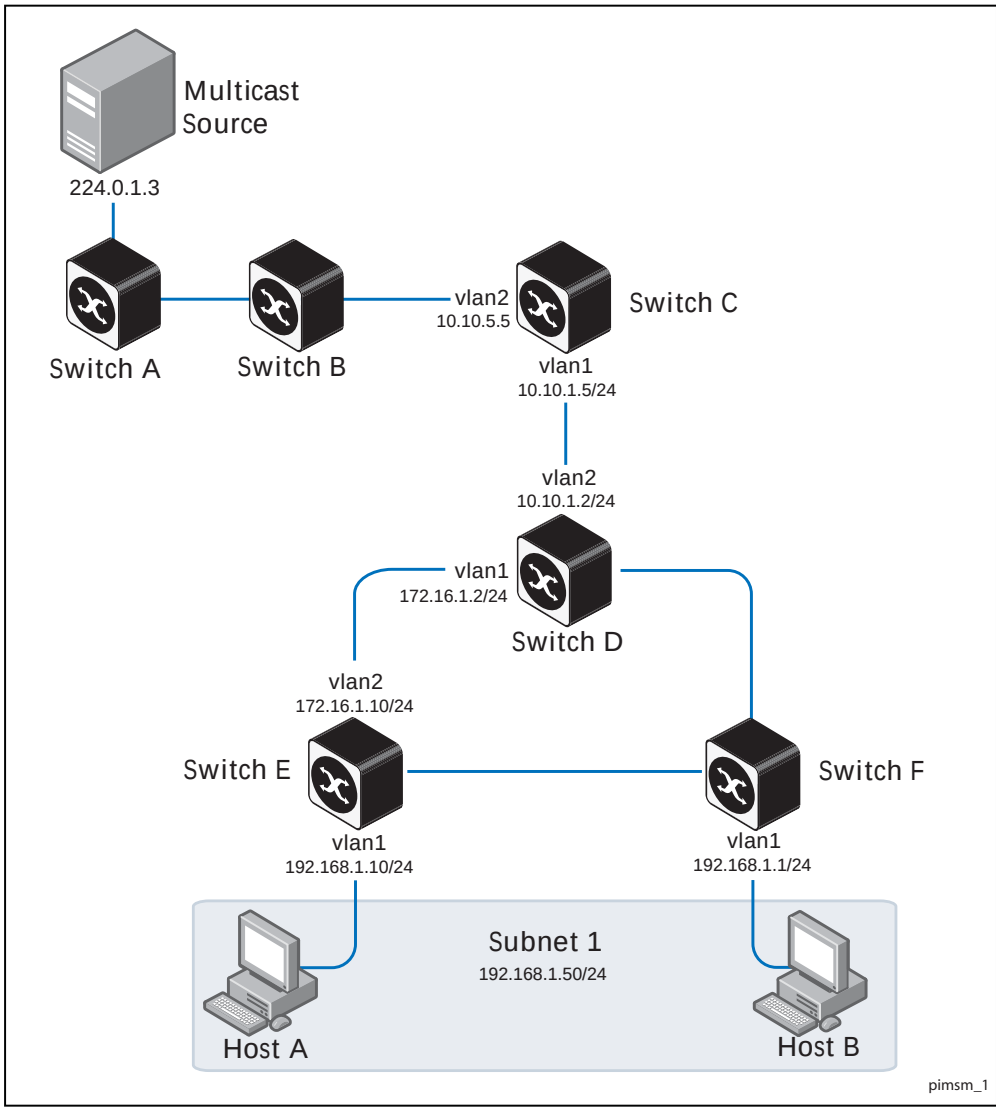
The switch that won the Assert election forwards these data packets, and acts as the local Designated Router for any IGMP members on the LAN. The downstream switches on the LAN also receive the Assert messages, and send all their join messages to the Assert winner. The result of an Assert election will timeout after the Assert Time. As long as the situation causing the duplication remains unchanged, the Assert winner sends an Assert message at a the Assert time interval, before the previous Assert messages time out. When the last downstream switch leaves the SPT, the Assert winner sends an Assert Cancel message saying that it is about to stop forwarding data on the SPT. Any RPT downstream switches then switch back to the RP tree.

PIM-SM Configuration

This section provides three PIM-SM configuration examples:

- **Static Rendezvous Point configuration**
- **Dynamic Rendezvous Point Configuration**
- **Bootstrap Router Configuration**

Both Rendezvous Point (RP) configuration examples refer to the network topology in the following graphic and use Allied Telesis managed Layer 3 Switches as the PIM routers. For details on the commands used in the following examples, refer to **Chapter 51, PIM-SM Commands**.



Static Rendezvous Point configuration

In this example using the above network topology, Switch C is the Rendezvous Point (RP) and all switches are statically configured with RP information. Host A and Host B join group `224.0.1.3` for all the sources. They send the IGMP membership report to Subnet 1. Two switches are attached to Subnet 1, Switch E and Switch F. Both of these switches have default Designated Router (DR) priority on `vlan1`. Because Switch E has a higher IP address on `vlan1`, Switch E becomes the DR and is responsible for sending Join messages to the RP (Switch C).

While configuring the RP, ensure that:

- Every switch includes the **ip pim rp-address 10.10.1.5** statement, even if it does not have any source or group member attached to it.
- There is only one RP address for the whole multicast group.
- All interfaces running PIM-SM must have sparse-mode enabled. In the configuration sample output below, both **vlan1** and **vlan2** are pim sparse-mode enabled.

See the following configuration output for **Switch D**:

```
hostname Switch D
!!
interface vlan1
 ip pim sparse-mode
!
interface vlan2
 ip pim sparse-mode
!
interface lo
!
!
ip multicast-routing
ip pim rp-address 10.10.1.5
!
```

Configure all the switches with the same **ip pim rp-address 10.10.1.5** statement as shown above.

Verifying configuration

Use the following commands to verify the RP configuration, interface details, and the multicast routing table.

RP details

For **Switch D**, the **show ip pim sparse-mode rp mapping** command shows that `10.10.1.5` is the RP for all multicast groups `224.0.0.0/4`, and is statically configured. All other switches will have a similar output.

```
awplus#show ip pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): 224.0.0.0/4, Static
    RP: 10.10.1.5
         Uptime: 5d02h15m
```

For **Switch D**, the **show ip pim sparse-mode rp-hash** command displays the selected RP for the specified group, in this example `224.0.1.3`.

```
awplus#show ip pim sparse-mode rp-hash 224.0.1.3
    RP: 10.10.1.5
```

Interface details

For **Switch E**, the **show ip pim sparse-mode interface** command displays the interface details and shows that Switch E is the DR on Subnet 1.

```
awplus#show ip pim sparse-mode interface
Total configured interfaces: 16   Maximum allowed: 31
Total active interfaces:     12

Address      Interface VIFindex Ver/   Nbr      DR    DR
                                       Mode    Count  Prior
192.168.1.10    vlan2  0       v2/S    1          1   192.168.1.10
172.16.1.10     vlan3  2       v2/S    1          1   172.16.1.10
```

IP multicast routing table

Note that the multicast routing table displayed for an RP switch is different to that displayed for other switches. For **Switch C**, because this switch is the RP and the root of this multicast tree, the **show ip pim sparse-mode mroute** command shows **RPF nbr** (next-hop to reach RP) as **0.0.0.0** and **RPF idx** (incoming interface for this (*, G) state) as **None**.

```
awplus#show ip pim sparse-mode mroute
IP Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
(*, 224.0.1.3)
RP: 10.10.1.5
RPF nbr: 0.0.0.0
RPF idx: None
Upstream State: JOINED
 Local     ..............................
 Joined    j..............................
 Asserted  ..............................
 Outgoing  o..............................
```

For **Switch E**, the **show ip pim sparse-mode mroute** command displays the IP multicast routing table.

```
awplus#show ip pim sparse-mode mroute
IP Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
(*, 224.0.1.3)
RP: 10.10.1.5
RPF nbr: 172.16.1.2
Upstream State: JOINED
 Local     ..............................
 Joined    j..............................
 Asserted  ..............................
 Outgoing  o..............................
```

On Switch E, `port1.0.2` is the incoming interface of the (*, G) entry, and `port1.0.1` is on the outgoing interface list of the (*, G) entry. This means that there is a group member through `port1.0.1`, and RP is reachable through `port1.0.2`.

Dynamic Rendezvous Point Configuration

A static RP configuration works for a small, stable PIM domain. However, it is not practical for a large and not so stable internetwork. In such a network, if the RP fails, the network administrator may have to change the static configurations on all PIM switches. An additional reason for choosing dynamic configuration high routing traffic leading to a change in the RP.

The Bootstrap Router (BSR) mechanism is used to dynamically maintain the RP information. To configure the RP dynamically in the above network topology, Switch C on `port1.0.1` and Switch D on `vlan1` are configured as RP candidates using the **ip pim rp-candidate** command. Switch D on `vlan1` is also configured as the BSR candidate. Since no other device has been configured as a BSR candidate, Switch D becomes the BSR router and is responsible for sending group-to-RP mapping information to all other PIM switches in this PIM domain.

The following output displays the complete configuration at **Switch C**.

```
awplus#show run
!
interface vlan1
 ip pim sparse-mode
!
interface vlan2
 ip pim sparse-mode
!
interface lo
!
ip multicast-routing
ip pim rp-candidate vlan1
```

The following output displays the complete configuration at **Switch D**.

```
awplus#show run
!
interface vlan1
 ip pim sparse-mode
!
interface vlan2
 ip pim sparse-mode
!
interface lo
!
ip multicast-routing
ip pim bsr-candidate vlan1
ip pim rp-candidate vlan1 priority 2
!
```

The highest priority switch is chosen as the RP. If two or more switches have the same priority, a hash function in the BSR mechanism is used to choose the RP to make sure that all devices in the PIM domain have the same RP for the same multicast group.

Use the `<interface> [priority <priority>]` parameters of the **ip pim rp-candidate** command to change the default priority of any RP candidate.

PIM group-to-RP mappings

The **show ip pim sparse-mode rp mapping** command displays the group-to-RP mapping details. The output shows information about RP candidates. There are two RP candidates for the group range `224.0.0.0/4`. RP candidate `10.10.1.5` has a default priority of `192`, whereas RP candidate `172.16.1.2` has been configured to have a priority of `2`. Since RP candidate `172.16.1.2` has a higher priority, it is selected as the RP for the multicast group `224.0.0.0/4`.

See the following configuration output for **Switch D**.

```
awplus#show ip pim sparse-mode rp mapping
This system is the Bootstrap Router (v2)
Group(s): 224.0.0.0/4
  RP: 10.10.1.5
    Info source: 172.16.1.2, via bootstrap, priority 192
         Uptime: 00:00:13, expires: 00:02:29
  RP: 172.16.1.2
    Info source: 172.16.1.2, via bootstrap, priority 2
         Uptime: 00:34:42, expires: 00:01:49
```

RP details

The **show ip pim sparse-mode rp-hash** command displays information about the RP router for a particular group. See the following configuration output for **Switch D**. This output shows that `172.16.1.2` has been chosen as the RP for the multicast group `224.0.1.3`.

```
awplus#show ip pim sparse-mode rp-hash 224.0.1.3
Group(s): 224.0.0.0/4
    RP: 172.16.1.2
    Info source: 172.16.1.2, via bootstrap
```

After RP information reaches all PIM switches in the domain, various state machines maintain all routing states as the result of Join/Prune messages from members of the multicast group.

Bootstrap Router Configuration

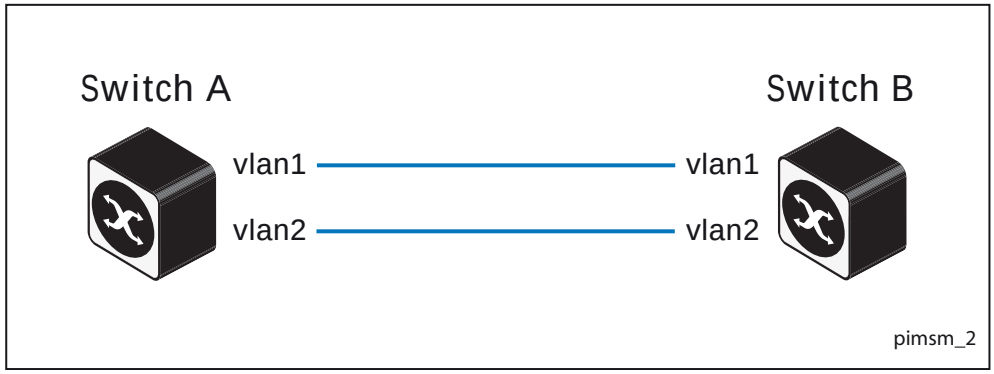
Every PIM multicast group needs to be associated with the IP address of a Rendezvous Point (RP). This address is used as the root of a group-specific distribution tree, whose branches extend to all nodes in the domain that want to receive traffic sent to the group. For all senders to reach all receivers, all devices in the domain use the same mappings of group addresses to RP addresses. In order to determine the RP for a multicast group, a PIM device maintains a collection of group-to-RP mappings, called the RP-Set.

The Bootstrap Router (BSR) mechanism for the class of multicast routing protocols in the PIM domain uses the concept of an RP as a means for receivers to discover the sources that send to a particular multicast group. The BSR mechanism is one way that a multicast router can learn the set of group-to-RP mappings required in order to function.

Some of the PIM devices within a PIM domain are configured as RP candidates. A subset of the RP candidates will eventually be used as the actual RPs for the domain. An RP configured with a lower value in the priority field has higher a priority.

Some of the PIM devices in the domain are configured to be BSR candidates. One of these BSR candidates is elected to be the BSR for the domain, and all PIM devices in the domain learn the result of this election through Bootstrap messages (BSM). The BSR candidate with highest value in the priority field is the elected BSR.

The RP candidates then report their candidacy to the elected BSR, which chooses a subset of the RP candidates, and distributes corresponding group-to-RP mappings to all the devices in the domain through Bootstrap messages.



Switch A Enter the following commands to configure `vlan1` on Switch A as the BSR candidate. The default priority is 64.

```
awplus# configure terminal
awplus(config)# ip pim bsr-candidate vlan1
awplus(config)# exit
```

Switch B Enter the following commands to configure `vlan1` on Switch B as the BSR candidate with a hash mask length of 10 and a priority of 25 and to configure `vlan1` as the RP candidate with a priority of 0.

```
awplus# configure terminal
awplus(config)# ip pim bsr-candidate vlan1 10 25
awplus(config)# ip pim rp-candidate vlan1 priority 0
awplus(config)# exit
```

Validation commands

Use the **show ip pim sparse-mode bsr-router** command to verify the BSR candidate state on **Switch A**.

```
awplus#show ip pim sparse-mode bsr-router
PIMv2 Bootstrap information
This system is the Bootstrap Router (BSR)
  BSR address: 20.0.1.21
  Uptime:      00:37:12, BSR Priority: 64, Hash mask length: 10
  Expires:     00:01:32
  Role: Candidate BSR
  State: Elected BSR
```

Use the **show ip pim sparse-mode bsr-router** command to verify the BSR candidate state on **Switch B**. The initial state of the BSR candidate is pending before transitioning to BSR candidate.

```
awplus#show ip pim sparse-mode bsr-router
PIMv2 Bootstrap information
  BSR address: 20.0.1.21
  Uptime:      00:02:39, BSR Priority: 64, Hash mask length: 10
  Expires:     00:00:03
  Role: Candidate BSR
  State: Pending BSR

awplus#show ip pim sparse-mode bsr-router
PIMv2 Bootstrap information
 BSR address: 20.0.1.21
 Uptime:      00:40:20, BSR Priority: 64, Hash mask length: 10
 Expires:     00:02:07
 Role: Candidate BSR
 State: Candidate BSR
```

Use the **show ip pim sparse-mode rp mapping** command to verify RP-set information on **Switch A**.

```
awplus#show ip pim sparse-mode rp mapping
PIM Group-to-RP Mappings
This system is the Bootstrap Router (v2)
Group(s): 224.0.0.0/4
 RP: 20.0.1.11
 Info source: 20.0.1.11, via bootstrap, priority 0
 Uptime: 00:00:30, expires: 00:02:04
```

Use the **show ip pim sparse-mode rp mapping** command to verify RP-set information on **Switch B**.

```
awplus#show ip pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): 224.0.0.0/4
 RP: 20.0.1.11
 Info source: 20.0.1.21, via bootstrap, priority 0
 Uptime: 00:00:12, expires: 00:02:18
```

PIM-SSM

Protocol Independent Multicast - Source Specific Multicast (PIM-SSM) is derived from Protocol Independent Multicast - Sparse Mode (PIM-SM) and is a simplified version of PIM-SM.

For details of the commands used to configure PIM-SSM, see **Chapter 51, PIM-SM Commands** and **Chapter 49, IGMP and IGMP Snooping Commands**.

Characteristics of PIM-SSM

While PIM-SM supports both a “many-to-many” and a “one-to-many” model, PIM-SSM only supports the “one-to-many” model, also known as a “broadcast application”. It is possible for devices in a network to support both PIM-SM and PIM-SSM at the same time, even on the same interfaces.

Instead of multicast groups being initially sent to a Rendezvous Point (RP), PIM-SSM builds shortest path trees (SPT) that are directly rooted at the source. This is because PIM-SSM relies on the multicast client informing the closest multicast router of the source IP address of the multicast server, as well as the multicast group IP address for the group it wants to join.

This approach removes some of the complexity in the multicast routing process and also improves security. Because a multicast client sends a join that includes the specific source IP address of the server it is no longer possible for an attacker to broadcast to the same group IP address from a different source IP address and cause a Denial of Service (DoS) attack. With PIM-SSM the attacker's traffic is discarded at the edge of the PIM domain.

PIM-SSM IP Address Range

The Internet Assigned Numbers Authority (IANA) has reserved the address range 232.0.0.0 through 232.255.255.255 for SSM applications. Although PIM-SSM can technically be configured to use the entire 224/4 multicast address range, PIM-SSM operation is guaranteed only in the 232/8 range (232.0.0/24 is reserved).

IGMPv3 and SSM-Mapping

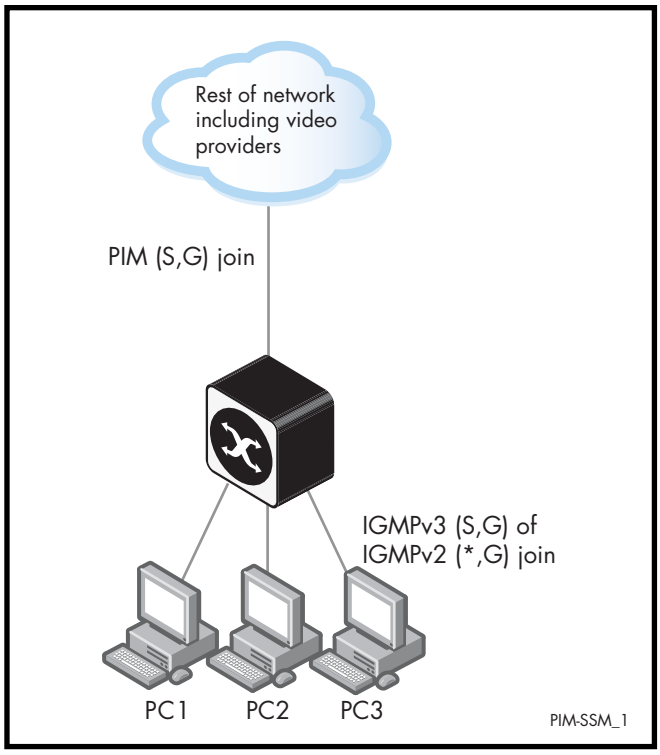
A restriction of PIM-SSM is that it requires a “Source, Group” (S,G) join and only IGMPv3 supports this. Source Specific Multicast Mapping is required to use PIM-SSM when you have older multicast client devices that do not support IGMPv3. This additional feature allows you to statically map IGMPv1/v2 (*,G) joins into PIM (S,G) joins, which in turn allows the device to talk to an upstream PIM-SSM network.

Note that IGMPv3 (*,G) joins cannot be mapped by SSM-Mapping.

How PIM-SSM Works

To join multicast group `232.1.1.1` each PC must send an IGMPv3 join with the source IP address specified. The join will be a (S,G) join, for example (`85.1.1.1,232.1.1.1`). The router will receive the IGMP join and check if the group address is in the SSM range. Then:

- If the group address is in the SSM range, the router will verify that a specific source or sources have been included in the IGMP join.
- If a specific source or sources has been included in the IGMP join, then the router will forward a PIM (S,G) join towards the source IP address.
- If the source IP address is not specified, then the router will discard the IGMP join and the PC will not join the group.
- If an IGMPv2 join is received for the SSM range then by default the join is discarded because no source IP address is specified. IGMP joins for group addresses that are not in the SSM range do not need to specify a specific source IP address.



How IGMP SSM-Mapping Works

In the example above (**"How PIM-SSM Works"**), if an IGMPv2 join is sent it is discarded because IGMPv2 only supports (*,G) joins. To resolve this issue, IGMP SSM-Mapping allows the router to be statically configured with source IP addresses for each group address or range of group addresses. This allows the router to receive a (*,G) join, match the group address via a software ACL, and based on this, insert the matching source IP address. The router then treats the join as a normal (S,G) join. If no match is found then the (*,G) is used. If the group address is in the SSM range then the join is discarded.

Configure PIM-SSM

Table 50-1: General configuration procedure for PIM-SSM

To enable SSM on the device	
`awplus#` `configure terminal`	Enter Global Configuration mode.
`awplus(config)#` `ip igmp ssm-map enable`	This command applies to VLAN interfaces configured for IGMP.
To specify the static mode of defining SSM	
`awplus#` `configure terminal`	Enter Global Configuration mode.

`awplus(config)#`
`access-list {<1-99>|<1300-1999>} permit <source>`
OR
`access-list standard <standard-access-list-name> permit <source>`
	Configure either a Standard Numbered ACL, Expanded Numbered ACL, or Standard Named ACL. Specify a multicast group address range and wildcard mask with the *source* parameter.

`awplus(config)#` `ip igmp ssm-map static {<access-list-number>|<access-list-number-expanded>|<access-list-name>} <ip-address>`	This command applies to VLAN interfaces configured for IGMP. SSM statically assigns sources to IGMPv1 and IGMPv2 groups to translate such (*,G) groups' memberships to (S,G) memberships for use with PIM-SSM.
To define a non-default SSM range of IP multicast addresses in IGMP	
`awplus#` `configure terminal`	Enter Global Configuration mode.

`awplus(config)#`
`access-list {<1-99>} permit <source>`
OR
`access-list standard <standard-access-list-name> permit <source>`
	Configure either a Standard Numbered ACL or Standard Named ACL.

Table 50-1: General configuration procedure for PIM-SSM (cont.)

`awplus(config)#`	
`ip igmp ssm range {<access-list-number>|<access-list-name>}`	Incoming IGMPv1 and IGMPv2 join requests are ignored if the multicast IP address is in the SSM range and no SSM mapping is configured for these addresses. By default, the SSM range is 232/8. To define the SSM range to be other than the default, specify either an access-list name or and access-list number.
To define the Source Specific Multicast (SSM) range of IP multicast addresses	
`awplus#`	
	Enter Global Configuration mode.
`awplus(config)#`	
`ip pim ssm default`	The default keyword defines the SSM range as 232/8.
	OR
`awplus(config)#`	
`ip pim ssm range {<access-list>|<named-access-list>}`	To define the SSM range to be other than the default, use the access-list parameter option.

Chapter 51: PIM-SM Commands

Command List 51.2
clear ip pim sparse-mode bsr rp-set * 51.2
clear ip mroute pim sparse-mode 51.3
debug pim sparse-mode 51.4
debug pim sparse-mode timer 51.5
ip pim accept-register list 51.7
ip pim anycast-rp 51.8
ip pim bsr-border 51.9
ip pim bsr-candidate 51.10
ip pim cisco-register-checksum 51.11
ip pim cisco-register-checksum group-list 51.12
ip pim crp-cisco-prefix 51.13
ip pim dr-priority 51.14
ip pim exclude-genid 51.15
ip pim ext-srcs-directly-connected (PIM-SM) 51.16
ip pim hello-holdtime (PIM-SM) 51.17
ip pim hello-interval (PIM-SM) 51.18
ip pim ignore-rp-set-priority 51.19
ip pim jp-timer 51.20
ip pim neighbor-filter (PIM-SM) 51.21
ip pim register-rate-limit 51.22
ip pim register-rp-reachability 51.22
ip pim register-source 51.23
ip pim register-suppression 51.24
ip pim rp-address 51.25
ip pim rp-candidate 51.27
ip pim rp-register-kat 51.28
ip pim sparse-mode 51.29
ip pim sparse-mode passive 51.30
ip pim spt-threshold 51.31
ip pim spt-threshold group-list 51.32
ip pim ssm 51.33
show debugging pim sparse-mode 51.34
show ip pim sparse-mode bsr-router 51.34
show ip pim sparse-mode interface 51.35
show ip pim sparse-mode interface detail 51.36
show ip pim sparse-mode local-members 51.37
show ip pim sparse-mode mroute 51.38
show ip pim sparse-mode mroute detail 51.40
show ip pim sparse-mode neighbor 51.42
show ip pim sparse-mode nexthop 51.43
show ip pim sparse-mode rp-hash 51.44
show ip pim sparse-mode rp mapping 51.44
undebug all pim sparse-mode 51.45

Command List

This chapter provides an alphabetical reference of PIM-SM commands. For commands common to PIM-SM and PIM-DM, see **Chapter 47, Multicast Introduction and Commands**.

clear ip pim sparse-mode bsr rp-set *

Use this command to clear all Rendezvous Point (RP) sets learned through the PIMv2 Bootstrap Router (BSR).

Syntax

```
clear ip pim sparse-mode bsr rp-set *
```

Parameter	Description
*	Clears all RP sets.

Mode Privileged Exec

Usage For multicast clients, note that one router will be automatically or statically designated as the RP, and all routers must explicitly join through the RP. A Designated Router (DR) sends periodic Join/Prune messages toward a group-specific RP for each group that it has active members.

For multicast sources, note that the Designated Router (DR) unicasts Register messages to the RP encapsulating the data packets from the multicast source. The RP forwards decapsulated data packets toward group members.

Example

```
awplus# clear ip pim sparse-mode bsr rp-set *
```

clear ip mroute pim sparse-mode

Use this command to clear all multicast route table entries learned through PIM-SM for a specified multicast group address, and optionally a specified multicast source address.

Syntax

```
clear ip mroute <Group-IP-address> pim sparse-mode
clear ip mroute <Group-IP-address> <Source-IP-address> pim sparse-mode
```

Parameter	Description
<Group-IP-address>	Specify a multicast group IPv6 address, entered in the form `A.B.C.D`.
<Source-IP-address>	Specify a source group IP address, entered in the form `A.B.C.D`.

Mode Privileged Exec

Example

```
awplus# clear ip mroute 224.0.0.0 pim sparse-mode

awplus# clear ip mroute 224.0.0.0 192.168.7.1 pim
       sparse-mode
```

debug pim sparse-mode

Use this command to activate/de-activate all PIM-SM debugging.

Syntax

```
debug pim sparse-mode [all] [events] [mfc] [mib] [nexthop] [nsm]
    [packet] [state] [mtrace]

no debug pim sparse-mode [all] [events] [mfc] [mib] [nexthop] [nsm]
    [packet] [state] [mtrace]
```

Parameter	Description
`all`	Activates/deactivates all PIM-SM debugging.
`events`	Activates debug printing of events.
`mfc`	Activates debug printing of MFC (Multicast Forwarding Cache in kernel) add/delete/updates.
`mib`	Activates debug printing of PIM-SM MIBs.
`nexthop`	Activates debug printing of PIM-SM next hop communications.
`nsm`	Activates debugging of PIM-SM Network Services Module communications.
`packet`	Activates debug printing of incoming and/or outgoing packets.
`state`	Activates debug printing of state transition on all PIM-SM FSMs.
`mtrace`	Activates debug printing of multicast traceroute.

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim sparse-mode all
```

Related Commands **show debugging pim sparse-mode**
undebug all pim sparse-mode

debug pim sparse-mode timer

Use this command to enable debugging for the specified PIM-SM timers.

Use the **no** variants of this command to disable debugging for the specified PIM-SM timers.

Syntax

```
debug pim sparse-mode timer assert [at]
no debug pim sparse-mode timer assert [at]
debug pim sparse-mode timer bsr [bst|crp]
no debug pim sparse-mode timer bsr [bst|crp]
debug pim sparse-mode timer hello [ht|nlt|tht]
no debug pim sparse-mode timer hello [ht|nlt|tht]
debug pim sparse-mode timer joinprune [jt|et|ppt|kat|ot]
no debug pim sparse-mode timer joinprune [jt|et|ppt|kat|ot]
debug pim sparse-mode timer register [rst]
no debug pim sparse-mode timer register [rst]
```

Parameter	Description
`assert`	Enable or disable debugging for the Assert timers.
`at`	Enable or disable debugging for the Assert Timer.
`bsr`	Enable or disable debugging for the specified Bootstrap Router timer, or all Bootstrap Router timers.
`bst`	Enable or disable debugging for the Bootstrap Router: Bootstrap Timer.
`crp`	Enable or disable debugging for the Bootstrap Router: Candidate-RP Timer.
`hello`	Enable or disable debugging for the specified Hello timer, or all Hello timers.
`ht`	Enable or disable debugging for the Hello timer: Hello Timer.
`nlt`	Enable or disable debugging for the Hello timer: Neighbor Liveness Timer.
`tht`	Enable or disable debugging for the Hello timer: Triggered Hello Timer.
`joinprune`	Enable or disable debugging for the specified JoinPrune timer, or all JoinPrune timers.
`jt`	Enable or disable debugging for the JoinPrune timer: upstream Join Timer.
`et`	Enable or disable debugging for the JoinPrune timer: Expiry Timer.
`ppt`	Enable or disable debugging for the JoinPrune timer: PrunePending Timer.
`kat`	Enable or disable debugging for the JoinPrune timer: KeepAlive Timer.

Parameter	Description
`ot`	Enable or disable debugging for the JoinPrune timer: Upstream Override Timer.
`register`	Enable or disable debugging for the Register timers.
`rst`	Enable or disable debugging for the Register timer: Register Stop Timer.

Default By default, all debugging is disabled.

Mode Privileged Exec and Global Configuration

Examples To enable debugging for the PIM-SM Bootstrap Router bootstrap timer, use the commands:

```
awplus(config)# debug pim sparse-mode timer bsr bst
```

To enable debugging for the PIM-SM Hello: neighbor liveness timer, use the command:

```
awplus(config)# debug pim sparse-mode timer hello ht
```

To enable debugging for the PIM-SM Joinprune expiry timer, use the command:

```
awplus# debug pim sparse-mode timer joinprune et
```

To disable debugging for the PIM-SM Register timer, use the command:

```
awplus# no debug pim sparse-mode timer register
```

Related Commands **show debugging pim sparse-mode**

ip pim accept-register list

Use this command to configure the ability to filter out multicast sources specified by the given access-list at the Rendezvous Point (RP), so that the RP will accept/refuse to perform the register mechanism for the packets sent by the specified sources. By default, the RP accepts register packets from all multicast sources.

Use the **no** variant of this command to revert to default.

Syntax

```
ip pim accept-register list{<simplerange>|<exprange>|<access-list>}
no ip pim accept-register
```

Parameter	Description
<simplerange>	<100-199> IP extended access-list.
<exprange>	<2000-2699> IP extended access list (expanded range).
<access-list>	IP Named Standard Access list.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim accept-register list 121
awplus(config)# access-list 121 permit ip 100.1.1.1 0.0.0.0 any
```

ip pim anycast-rp

Use this command to configure Anycast RP (Rendezvous Point) in a RP set.

Use the **no** variant of this command to remove the configuration.

Syntax

```
ip pim anycast-rp <anycast-rp-address> <member-rp-address>
no ip pim anycast-rp <anycast-rp-address> [<member-rp-address>]
```

Parameter	Description
<anycast-rp-address>	<A.B.C.D> Specify an anycast IP address to configure an Anycast RP (Rendezvous Point) in a RP set.
<member-rp-address>	<A.B.C.D> Specify an Anycast RP (Rendezvous Point) address to configure an Anycast RP in a RP set.

Mode Global Configuration

Usage Anycast is a network addressing and routing scheme where data is routed to the nearest or best destination as viewed by the routing topology. Compared to unicast with a one-to-one association between network address and network endpoint, and multicast with a one-to-many association between network address and network endpoint; anycast has a one-to-many association between network address and network endpoint. For anycast, each destination address identifies a set of receiver endpoints, from which only one receiver endpoint is chosen.

Use this command to specify the Anycast RP configuration in the Anycast RP set. Use the **no** variant of this command to remove the Anycast RP configuration. Note that the member RP address is optional when using the **no** parameter to remove the Anycast RP configuration. removing the anycast RP address also removes the member RP address.

Examples The following example shows how to configure the Anycast RP address with **ip pim anycast-rp**:

```
awplus# configure terminal
awplus(config)# ip pim anycast-rp 1.1.1.1 10.10.10.10
```

The following example shows how to remove the Anycast RP in the RP set specifying only the anycast RP address with **no ip pim anycast-rp**, but not specifying the member RP address:

```
awplus# configure terminal
awplus(config)# no ip pim anycast-rp 1.1.1.1
```

ip pim bsr-border

Use the **ip pim bsr-border** command to prevent Bootstrap Router (BSR) messages from being sent or received through a VLAN interface. The BSR border is the border of the PIM domain.

Use the **no** variant of this command to disable the configuration set with **ip pim bsr-border**.

Syntax

```
ip pim bsr-border
no ip pim bsr-border
```

Mode Interface Configuration for a VLAN interface.

Usage When this command is configured on a VLAN interface, no PIM version 2 BSR messages will be sent or received through the interface. Configure an interface bordering another PIM domain with this command to avoid BSR messages from being exchanged between the two PIM domains.

BSR messages should not be exchanged between different domains, because devices in one domain may elect Rendezvous Points (RPs) in the other domain, resulting in loss of isolation between the two PIM domains that would stop the PIM protocol from working as intended.

Examples The following example configures the VLAN interface `vlan2` to be the PIM domain border:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim bsr-border
```

The following example removes the VLAN interface `vlan2` from the PIM domain border:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim bsr-border
```

ip pim bsr-candidate

Use this command to give the device the candidate BSR (Bootstrap Router) status using the specified IP address mask of the interface.

Use the **no** variant of this command to withdraw the address of the interface from being offered as a BSR candidate.

Syntax

```
ip pim bsr-candidate <interface> [<hash>] [<priority>]
no ip pim bsr-candidate [<interface>]
```

Parameter	Description
<interface>	The interface. For instance, `vlan2`.
<hash>	`<0-32>` configure hash mask length for RP selection. The default hash value if you do not configure this parameter is 10.
<priority>	`<0-255>` configure priority for a BSR candidate. Note that you must also specify the <*hash*> (mask length) when specifying the <*priority*>. The default priority if you do not configure this parameter is 64.

Mode Global Configuration

Default The default hash parameter value is 10 and the default priority parameter value is 64.

Examples To set the BSR candidate to the VLAN interface `vlan2`, with the optional mask length and BSR priority parameters, enter the commands shown below:

```
awplus# configure terminal
awplus(config)# ip pim bsr-candidate vlan2 20 30
```

To withdraw the address of `vlan2` from being offered as a BSR candidate, enter:

```
awplus# configure terminal
awplus(config)# no ip pim bsr-candidate vlan2
```

ip pim cisco-register-checksum

Use this command to configure the option to calculate the Register checksum over the whole packet. This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to disable this option.

Syntax

```
ip pim cisco-register-checksum
no ip pim cisco-register-checksum
```

Default This command is disabled by default. By default, Register Checksum is calculated only over the header.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim cisco-register-checksum
```

ip pim cisco-register-checksum group-list

Use this command to configure the option to calculate the Register checksum over the whole packet on multicast groups specified by the access-list. This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to revert to default settings.

Syntax

```
ip pim cisco-register-checksum group-list [<simplerange>|<exprange>|
    <access-list>]

no ip pim cisco-register-checksum group-list [<simplerange>|
    <exprange>|<access-ist>]
```

Parameter	Description
<simplerange>	<1-99> Simple access-list.
<exprange>	<1300-1999> Simple access-list (expanded range).
<access-list>	IP Named Standard Access list.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim cisco-register-checksum group-list 34
awplus(config)# access-list 34 permit 224.0.1.3
```

ip pim crp-cisco-prefix

Use this command to interoperate with Cisco devices that conform to an earlier draft standard. Some Cisco devices might not accept candidate RPs with a group prefix number of zero. Note that the latest BSR specification prohibits sending RP advertisements with prefix 0. RP advertisements for the default IPv4 multicast group range 224/4 are sent with a prefix of 1.

Use the **no** variant of this command to revert to the default settings.

Syntax

```
ip pim crp-cisco-prefix
no ip pim crp-cisco-prefix
```

Mode Global Configuration

Usage Cisco's BSR code does not conform to the latest BSR draft. It does not accept candidate RPs with a group prefix number of zero. To make the candidate RP work with a Cisco BSR, use the **ip pim crp-cisco-prefix** command when interoperating with older versions of Cisco IOS.

Example

```
awplus# configure terminal
awplus(config)# ip pim crp-cisco-prefix

awplus# configure terminal
awplus(config)# no ip pim crp-cisco-prefix
```

Related Commands ip pim rp-candidate

ip pim dr-priority

Use this command to set the Designated Router priority value.

Use the **no** variant of this command to disable this function.

Syntax

```
ip pim dr-priority  <priority>
no ip pim dr-priority [<priority>]
```

Parameter	Description
<priority>	<0-4294967294> The Designated Router priority value. A higher value has a higher preference.

Default The default is 1. The negated form of this command restores the value to the default.

Mode Interface Configuration for a VLAN interface.

Examples To set the Designated Router priority value to `11234` for the VLAN interface `vlan2`, apply the commands as shown below:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim dr-priority 11234
```

To disable the Designated Router priority value for the VLAN interface `vlan2`, apply the commands as shown below:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim dr-priority
```

Related Commands ip pim ignore-rp-set-priority

ip pim exclude-genid

Use this command to exclude the GenID option from Hello packets sent out by the PIM module on a particular interface.This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to revert to default settings.

Syntax

```
ip pim exclude-genid
no ip pim exclude-genid
```

Default By default, this command is disabled; the GenID option is included.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim exclude-genid
```

ip pim ext-srcs-directly-connected (PIM-SM)

Use this command to configure PIM to treat all source traffic arriving on the interface as though it was sent from a host directly connected to the interface.

Use the **no** variant of this command to configure PIM to treat only directly connected sources as directly connected.

Syntax

```
ip pim ext-srcs-directly-connected
no ip pim ext-srcs-directly-connected
```

Default The **no** variant of this command is the default behavior.

Mode Interface Configuration for a VLAN interface.

Example To configure PIM to treat all sources as directly connected for VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim ext-srcs-directly-connected
```

To configure PIM to treat only directly connected sources as directly connected for VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim ext-srcs-directly-connected
```

ip pim hello-holdtime (PIM-SM)

This command configures a hello-holdtime value. You cannot configure a hello-holdtime value that is less than the current hello-interval.

Use the **no** variant of this command to return it to its default of 3.5 * the current hello-interval.

Syntax

```
ip pim hello-holdtime <holdtime>
no ip pim hello-holdtime
```

Parameter	Description
<holdtime>	<1-65535> The holdtime value in seconds (no fractional seconds are accepted).

Default The default hello-holdtime value is 3.5 * the current hello-interval. The default hello-holdtime is restored using the negated form of this command.

Mode Interface Configuration for a VLAN interface.

Usage Each time the hello interval is updated, the hello holdtime is also updated, according to the following rules:

If the hello holdtime is not configured; or if the hello holdtime is configured and less than the current hello-interval value, it is modified to the (3.5 * hello interval). Otherwise, it retains the configured value.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim hello-holdtime 123
```

ip pim hello-interval (PIM-SM)

This command configures a hello-interval value.

Use the **no** variant of this command to reset the hello-interval to the default.

Syntax

```
ip pim hello-interval <interval>
no ip pim hello-interval
```

Parameter	Description
<interval>	<1-65535> The value in seconds (no fractional seconds accepted).

Default The default hello-interval value is 30 seconds. The default is restored using the negated form of this command.

Mode Interface Configuration for a VLAN interface.

Usage When the hello interval is configured, and the hello holdtime is not configured, or when the configured hello-holdtime value is less than the new hello-interval value; the holdtime value is modified to the (3.5 * hello interval). Otherwise, the hello-holdtime value is the configured value.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim hello-interval 123
```

ip pim ignore-rp-set-priority

Use this command to ignore the RP-SET priority value, and use only the hashing mechanism for RP selection.

This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to disable this setting.

Syntax

```
ip pim ignore-rp-set-priority
no ip pim ignore-rp-set-priority
```

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim ignore-rp-set-priority
```

ip pim jp-timer

Use this command to set the PIM-SM join/prune timer. Note that the value the switch puts into the holdtime field of the join/prune packets it sends to its neighbors is 3.5 times the join/prune timer value set using this command.

Use the **no** variant of this command to return the PIM-SM join/prune timer to its default value of 60 seconds, which corresponds to a join/prune packet holdtime of 210 seconds.

Syntax

```
ip pim jp-timer <1-65535>
no ip pim jp-timer [<1-65535>]
```

Parameter	Description
<1-65535>	Specifies the join/prune timer value. The default value is 60 seconds.

Default The default join/prune timer value is 60 seconds.

Mode Global Configuration

Example To set the join/prune timer value to 300 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip pim jp-timer 300
```

To return the join/prune timer to its default value of 60 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# no ip pim jp-timer
```

ip pim neighbor-filter (PIM-SM)

This command enables filtering of neighbors on the VLAN interface. When configuring a neighbor filter, PIM-SM will either not establish adjacency with the neighbor, or terminate adjacency with the existing neighbors if denied by the filtering access list.

Use the **no** variant of this command to disable this function.

Syntax

```
ip pim neighbor-filter {<number>|<accesslist>}
no ip pim neighbor-filter {<number>|<accesslist>}
```

Parameter	Description
<number>	<1-99> Standard IP access-list number.
<accesslist>	IP access list name.

Default By default, there is no filtering.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim neighbor-filter 14
```

ip pim register-rate-limit

Use this command to configure the rate of register packets sent by this DR, in units of packets per second.

Use the **no** variant of this command to remove the limit.

Syntax

```
ip pim register-rate-limit <1-65535>
no ip pim register-rate-limit
```

Parameter	Description
<1-65535>	Specifies the maximum number of packets that can be sent per second.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim register-rate-limit 3444
```

ip pim register-rp-reachability

Use this command to enable the RP reachability check for PIM Register processing at the DR. The default setting is no checking for RP-reachability.

Use the **no** variant of this command to disable this processing.

Syntax

```
ip pim register-rp-reachability
no ip pim register-rp-reachability
```

Default This command is disabled; by default, there is no checking for RP-reachability.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim register-rp-reachability
```

ip pim register-source

Use this command to configure the source address of register packets sent by this DR, overriding the default source address, which is the address of the RPF interface toward the source host.

Use the **no** variant of this command to un-configure the source address of Register packets sent by this DR, reverting back to use the default source address that is the address of the RPF interface toward the source host.

Syntax

```
ip pim register-source [<sourceaddress>|<interface>]
no ip pim register-source
```

Parameter	Description
<sourceaddress>	The IP address, entered in the form `A.B.C.D`, to be used as the source of the register packets.
<interface>	The name of the interface to be used as the source of the register packets.

Usage The configured address must be a reachable address to be used by the RP to send corresponding Register-Stop messages in response. It is normally the local loopback interface address, but can also be a physical address. This address must be advertised by unicast routing protocols on the DR. The configured interface does not have to be PIM enabled.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim register-source 10.10.1.3
```

ip pim register-suppression

Use this command to configure the register-suppression time, in seconds, overriding the default of 60 seconds. Configuring this value modifies register-suppression time at the DR. Configuring this value at the RP modifies the RP-keepalive-period value if the **ip pim rp-register-kat** command on page 51.28 is not used.

Use the **no** variant of this command to reset the value to its default of 60 seconds.

Syntax

```
ip pim register-suppression <1-65535>
no ip pim register-suppression
```

Parameter	Description
<1-65535>	Register suppression on time in seconds.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ip pim register-suppression 192
```

ip pim rp-address

Use this command to statically configure RP (Rendezvous Point) address for multicast groups.

Use the **no** variant of this command to remove a statically configured RP (Rendezvous Point) address for multicast groups.

Syntax

```
ip pim rp-address <ip-address> [<simplerange>|<expandedrange>|
    <accesslist>] [<override>]

no ip pim rp-address <ip-address> [<simplerange>|<expandedrange>|
    <accesslist>] [override>]
```

Parameter	Description
<ip-address>	IP address of Rendezvous Point, entered in the form `A.B.C.D`.
<simplerange>	`<1-99>` IP Standard Access-list.
<expandedrange>	`<1300-1999>` IP Standard Access-list (expanded range).
<accesslist>	IP extended Access-list name.
<override>	Enables statically defined RPs to override dynamically learned RPs.

Mode Global Configuration

Usage The AlliedWare Plus™ PIM-SM implementation supports multiple static RPs. It also supports usage of static-RP and BSR mechanism simultaneously. The **ip pim rp-address** command is used to statically configure the RP address for multicast groups.

You need to understand the following information before using this command.

If the RP-address that is configured by the BSR, and the RP-address that is configured statically, are both available for a group range, then the RP-address configured through BSR is chosen over the statically configured RP-address.

A single static-RP can be configured for multiple group ranges using Access Lists. However, configuring multiple static RPs (using **ip pim rp-address** command) with the same RP address is not allowed. The static-RP can either be configured for the whole multicast group range `224.0.0.0/4` (without ACL) or for specific group ranges (using ACL).

For example, configuring **ip pim rp-address 192.168.3.4** will configure static-RP 192.168.3.4 for the default group range 224.0.0.0/4. Configuring i**p pim rp-address 192.168.7.8 grp-list** will configure static-RP 192.168.7.8 for all the group ranges represented by permit filters in grp-list ACL.

If multiple static-RPs are available for a group range, then one with the highest IP address is chosen.

Only `Permit` filters in ACL are considered as valid group ranges. The default `Permit` filter `0.0.0.0/0` is converted to the default multicast filter `224.0.0.0/4`.

After configuration, the RP-address is inserted into a static-RP group tree based on the configured group ranges. For each group range, multiple static-RPs are maintained in a linked list. This list is sorted in a descending order of IP addresses. When selecting static-RPs for a group range, the first element (which is the static-RP with highest IP address) is chosen.

RP-address deletion is handled by removing the static-RP from all the existing group ranges and recalculating the RPs for existing TIB states if required.

Group mode and RP address mappings learned through BSR take precedence over mappings statistically defined by the **ip pim rp-address** command. Commands with the **override** keyword take precedence over dynamically learned mappings.

Example

```
awplus# configure terminal
awplus(config)# ip pim rp-address 192.168.3.4 4
```

Related Commands

ip pim rp-candidate
ip pim rp-register-kat

ip pim rp-candidate

Use this command to give the router the candidate RP (Rendezvous Point) status using the IP address of the specified interface.

Use the **no** variant of this command to remove the RP status set using the **ip pim rp-candidate** command.

Syntax

```
ip pim rp-candidate <interface> [priority <priority>|interval
   <interval>| grouplist <grouplist>]

no ip pim rp-candidate [<interface>]
```

Parameter	Description
<interface>	Interface name
<priority>	<0-255> configure priority for an RP candidate.
<interval>	advertisement interval specified in the range <1-16383> (in seconds).
<grouplist>	IP access list specifier for standard, expanded or named access lists in their respective ranges: [<1-99>|*WORD*]

Default The priority value for a candidate RP is 192 by default until specified using the **priority** parameter.

Mode Global Configuration

Usage Note that issuing the command **ip pim rp-candidate** *<interface>* without optional **priority**, **interval**, or **grouplist** parameters will configure the candidate RP with a priority value of 192.

Examples

```
awplus# configure terminal
awplus(config)# ip pim rp-candidate vlan2 priority 3
```

```
awplus# configure terminal
awplus(config)# ip pim rp-candidate vlan2 priority 3
               group-list 3
```

```
awplus# configure terminal
awplus(config)# no ip pim rp-candidate vlan2
```

Related Commands **ip pim rp-address**
ip pim rp-register-kat

ip pim rp-register-kat

Use this command to configure the Keep Alive Time (KAT) for (S,G) states at the RP (Rendezvous Point) to monitor PIM-SM Register packets.

Use the **no** variant of this command to return the PIM-SM KAT timer to its default value of 210 seconds.

Syntax

```
ip pim rp-register-kat <1-65535>
no ip pim rp-register-kat
```

Parameter	Description
<1-65536>	Specify the KAT timer in seconds. The default value is 210 seconds.

Mode Global Configuration

Default The default PIM-SM KAT timer value is 210 seconds.

Examples

```
awplus# configure terminal
awplus(config)# ip pim rp-register-kat 3454

awplus# configure terminal
awplus(config)# no ip pim rp-register-kat
```

Related Commands **ip pim rp-address**
ip pim rp-candidate

ip pim sparse-mode

Use this command to enable PIM-SM on the VLAN interface.

Use the **no** variant of this command to disable PIM-SM on the VLAN interface.

Syntax

```
ip pim sparse-mode
no ip pim sparse-mode
```

Mode Interface Configuration for a VLAN interface.

Examples

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim sparse-mode

awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim sparse-mode
```

ip pim sparse-mode passive

Use this command to enable and disable passive mode operation for local members on the VLAN interface.

Use the **no** variant of this command to disable passive mode operation for local members on the VLAN interface.

Syntax

```
ip pim sparse-mode passive
no ip pim sparse-mode passive
```

Mode Interface Configuration for a VLAN interface.

Usage Passive mode essentially stops PIM transactions on the interface, allowing only IGMP mechanism to be active. To turn off passive mode, use the **no ip pim sparse-mode passive** or the **ip pim sparse-mode** command. To turn off PIM activities on the VLAN interface, use the **no ip pim sparse-mode** command.

Examples

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim sparse-mode passive
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim sparse-mode passive
```

ip pim spt-threshold

This command turns on the ability for the last-hop PIM router to switch to SPT.

The **no** variant of this command turns off the ability for the last-hop PIM router to switch to SPT.

Note The switching to SPT happens either at the receiving of the first data packet, or not at all; it is not rate-based.

Syntax

```
ip pim spt-threshold
no ip pim spt-threshold
```

Mode Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# ip pim spt-threshold

awplus# configure terminal
awplus(config)# no ip pim spt-threshold
```

ip pim spt-threshold group-list

Use this command to turn on/off the ability for the last-hop PIM router to switch to SPT for multicast group addresses specified by the given access-list.

The switching to SPT happens either at the receiving of the first data packet, or not at all; it is not rate-based.

Use the **no** variant of this command to turn off switching to the SPT.

Syntax

```
ip pim spt-threshold group-list {<simplerange>|<expandedrange>|
   <named-accesslist>}
no ip pim spt-threshold group-list [<simplerange>|<expandedrange>|
   <named-accesslist>]
```

Parameter	Description
<simplerange>	<1-99> IP Standard Access-list.
<expandedrange>	<1300-1999> IP Standard Access-list (expanded range).
<named-accesslist>	IP Access-list name.

Mode Global Configuration

Usage Turn on/off the ability for the last-hop PIM router to switch to SPT for multicast group addresses specified by the given access-list.

Example

```
awplus# configure terminal
awplus(config)# ip pim spt-threshold group-list 1
awplus(config)# access-list 1 permit 224.0.1.3
```

ip pim ssm

Use this command to define the Source Specific Multicast (SSM) range of IP multicast addresses. The default keyword defines the SSM range as 232/8. To define the SSM range to be other than the default, use the access-list parameter option.

Use the **no** variant of this command to disable the SSM range.

Syntax

```
ip pim ssm default
ip pim ssm range {<access-list>|<named-access-list>}
no ip pim ssm
```

Parameter	Description
<access-list>	<1-99> Simple access-list.
<named-access-list>	Named Standard Access List.

Default By default, the command is disabled.

Mode Global Configuration

Usage When an SSM range of IP multicast addresses is defined by the ip pim ssm command, the no (*,G) or (S,G,rpt) state will be initiated for groups in the SSM range.

The messages corresponding to these states will not be accepted or originated in the SSM range.

Examples The following commands show how to configure SSM service for the IP address range defined by access list 10:

```
awplus# configure terminal
awplus(config)# access-list 10 permit 225.1.1.1
awplus(config)# ip pim ssm range 10
```

The following commands show how to disable PIM-SSM:

```
awplus# configure terminal
awplus(config)# no ip pim ssm
```

show debugging pim sparse-mode

This command displays the status of the debugging of the system.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show debugging pim sparse-mode`

Mode User Exec and Privileged Exec

Example To display PIM-SM debugging settings, use the command:

```
awplus# show debugging pim sparse-mode
```

Figure 51-1: Output from the show debugging pim sparse-mode command

```
Debugging status:
  PIM event debugging is on
  PIM Hello THT timer debugging is on
  PIM event debugging is on
  PIM MFC debugging is on
  PIM state debugging is on
  PIM packet debugging is on
  PIM incoming packet debugging is on
  PIM outgoing packet debugging is on
```

Related Commands **debug pim sparse-mode**

show ip pim sparse-mode bsr-router

Use this command to show the Bootstrap Router (BSR) (v2) address.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip pim sparse-mode bsr-router`

Mode User Exec and Privileged Exec

Output **Figure 51-2: Output from the show ip pim sparse-mode bsr-router command**

```
PIMv2 Bootstrap information
  BSR address: 10.10.11.35 (?)
  Uptime:      00:00:38, BSR Priority: 0, Hash mask length: 10
  Expires:     00:01:32
  Role: Non-candidate BSR
  State: Accept Preferred
```

Related Commands **show ip pim sparse-mode rp mapping**
show ip pim sparse-mode neighbor

show ip pim sparse-mode interface

Use this command to show PIM-SM interface information.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip pim sparse-mode interface`

Mode User Exec and Privileged Exec

Example To display information about PIM-SM interfaces, use the command:

```
awplus# show ip pim sparse-mode interface
```

Output

```
Total configured interfaces: 16   Maximum allowed: 31
Total active interfaces:     12

Address         Interface VIFindex Ver/   Nbr    DR          DR
                                   Mode   Count  Prior
192.168.1.53    vlan2     0        v2/S   2      2       192.168.1.53
192.168.10.53   vlan3     2        v2/S   0      2       192.168.10.53

... Note that this screen has been edited to remove any additional interfaces.
```

Table 51-1: Parameters in the output from the show ip pim sparse-mode interface command

Parameters	Description
`Total configured interfaces`	The number of configured PIM Sparse Mode interfaces.
`Maximum allowed`	The maximum number of PIM Sparse Mode interfaces that can be configured.
`Total active interfaces`	The number of active PIM Sparse Mode interfaces.
`Address`	Primary PIM-SM address.
`Interface`	Name of the PIM-SM interface.
`VIF Index`	The Virtual Interface index of the VLAN.
`Ver/Mode`	PIM version/Sparse mode.
`Nbr Count`	Neighbor count of the PIM-SM interface.
`DR Priority`	Designated Router priority.
`DR`	The IP address of the Designated Router.

Related Commands
ip pim sparse-mode
show ip pim sparse-mode rp mapping
show ip pim sparse-mode neighbor

show ip pim sparse-mode interface detail

Use this command to show detailed information on a PIM-SM interface.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip pim sparse-mode interface detail`

Mode User Exec and Privileged Exec

Output **Figure 51-3: Example output from the show ip pim sparse-mode interface detail command**

```
vlan3 (vif 3):
  Address 192.168.1.149, DR 192.168.1.149
  Hello period 30 seconds, Next Hello in 15 seconds
  Triggered Hello period 5 seconds
  Neighbors:
   192.168.1.22

vlan2 (vif 0):
  Address 10.10.11.149, DR 10.10.11.149
  Hello period 30 seconds, Next Hello in 18 seconds
  Triggered Hello period 5 seconds
  Neighbors:
   10.10.11.4
```

show ip pim sparse-mode local-members

Use this command to show detailed local member information on a VLAN interface configured for PIM-SM. If you do not specify a VLAN interface then detailed local member information is shown for all VLAN interfaces configured for PIM-SM.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode local-members [<interface>]`

Parameter	Description
<interface>	Optional Specify the interface. For instance, VLAN interface `vlan2`.

Mode User Exec and Privileged Exec

Example To show detailed PIM-SM information for all PIM-SM configured VLAN interfaces, use the command:

```
awplus# show ipv6 pim sparse-mode local-members
```

Output **Figure 51-4: Example output from the show ip pim sparse-mode local-members command**

```
awplus#show ip pim sparse-mode local-members
PIM Local membership information

vlan1:

    (*, 224.0.0.4) : Include

vlan203:

    (*, 223.0.0.3) : Include
```

Example To show detailed PIM-SMv6 information for the PIM-SM configured interface `vlan1`, use the command:

```
awplus# show ipv6 pim sparse-mode local-members vlan1
```

Output **Figure 51-5: Example output from the show ip pim sparse-mode local-members vlan1 command**

```
awplus#show ip pim sparse-mode local-members vlan1
PIM Local membership information

vlan1:

    (*, 224.0.0.4) : Include
```

show ip pim sparse-mode mroute

This command displays the IP multicast routing table, or the IP multicast routing table based on the specified address or addresses.

Two group addresses cannot be used simultaneously; two source addresses cannot be used simultaneously.

Note that when a feature license is enabled, the output for the **show ip pim sparse-mode mroute** command will only show 32 interfaces because of the terminal display width limit. Use the **show ip pim sparse-mode mroute detail** command to display detailed entries of the IP multicast routing table.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip pim sparse-mode mroute [<group-address>|<source-address>]
show ip pim sparse-mode mroute [<source-address> <group-address>]
show ip pim sparse-mode mroute [<group-address> <source-address>]
```

Parameter	Description
<group-address>	Group IP address, entered in the form A.B.C.D. Based on the group and source address, the output is the selected route if present in the multicast route tree.
<source-address>	Source IP address, entered in the form A.B.C.D. Based on the source and group address, the output is the selected route if present in the multicast route tree.

Mode User Exec and Privileged Exec

Usage Note that when a feature license is enabled, the output for **show ip pim sparse-mode mroute** command will only show 32 interfaces because of the terminal display width limit. Use the **show ip pim sparse-mode mroute detail** command to display detailed entries of the IP multicast routing table.

Examples

```
awplus# show ip pim sparse-mode mroute
awplus# show ip pim sparse-mode mroute 40.40.40.11
awplus# show ip pim sparse-mode mroute 235.0.0.1
awplus# show ip pim sparse-mode mroute 235.0.0.1 40.40.40.11
```

Figure 51-6: Example output from the show ip pim sparse-mode mroute command

```
IP Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 1

(*, 224.0.1.3)
RP: 10.10.5.153
RPF nbr: 192.168.1.152
RPF idx: vlan2
Upstream State: JOINED
 Local     ..............................
 Joined    ..j...........................
 Asserted  ..............................
FCR:
Source: 10.10.1.52
 Outgoing  ..o...........................
 KAT timer running, 144 seconds remaining
 Packet count 1
```

show ip pim sparse-mode mroute detail

This command displays detailed entries of the IP multicast routing table, or detailed entries of the IP multicast routing table based on the specified address or addresses.

Two group addresses cannot be used simultaneously; two source addresses cannot be used simultaneously.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip pim sparse-mode mroute [<group-address>|<source-address>]
   detail

show ip pim sparse-mode mroute [<group-address> <source-address>]
   detail

show ip pim sparse-mode mroute [<source-address> <group-address>]
   detail
```

Parameter	Description
<group-address>	Group IP address, entered in the form A.B.C.D. Output is all multicast entries belonging to that group.
<source-address>	Source IP address, entered in the form A.B.C.D. Output is all multicast entries belonging to that source.
`detail`	Show detailed information.

Usage

Based on the group and source address, the output is the selected route if present in the multicast route tree.

Mode

User Exec and Privileged Exec

Examples

```
awplus# show ip pim sparse-mode mroute detail

awplus# show ip pim sparse-mode mroute 40.40.40.11 detail

awplus# show ip pim sparse-mode mroute 224.1.1.1 detail

awplus# show ip pim sparse-mode mroute 224.1.1.1 40.40.40.11
       detail
```

Figure 51-7: Example output from the show ip pim sparse-mode mroute detail command

```
IP Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 4
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 0

(*, 224.0.1.24) Uptime: 00:06:42
  RP: 0.0.0.0, RPF nbr: None, RPF idx: None
  Upstream:
   State: JOINED, SPT Switch: Disabled, JT: off
   Macro state: Join Desired,
  Downstream:
   vlan2:
     State: NO INFO, ET: off, PPT: off
     Assert State: NO INFO, AT: off
      Winner: 0.0.0.0, Metric: 42949672951, Pref: 42949672951,
RPT bit: on
     Macro state: Could Assert, Assert Track
  Local Olist:
   vlan2
```

show ip pim sparse-mode neighbor

Use this command to show the PIM-SM neighbor information.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax

```
show ip pim sparse-mode neighbor [<interface>] [<ip-address>]
   [detail]
```

Parameter	Description
<interface>	Interface name (e.g. `vlan2`). Show neighbors on an interface.
<ip-address>	Show neighbors with a particular address on an interface. The IP address entered in the form A.B.C.D.
`detail`	Show detailed information.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ip pim sparse-mode neighbor
```

```
awplus# show ip pim sparse-mode neighbor vlan5 detail
```

Figure 51-8: Example output from the show ip pim sparse-mode neighbor command

```
Neighbor         Interface   Uptime/Expires    Ver   DR
Address                                              Priority/
Mode
10.10.0.9        vlan2       00:55:33/00:01:44 v2    1 /
10.10.0.136      vlan2       00:55:20/00:01:25 v2    1 /
10.10.0.172      vlan2       00:55:33/00:01:32 v2    1 / DR
192.168.0.100    vlan3       00:55:30/00:01:20 v2    N / DR
```

Figure 51-9: Example output from the show ip pim sparse-mode neighbor interface detail command

```
Nbr 10.10.3.180 (vlan5), DR
 Expires in 55 seconds, uptime 00:00:15
 Holdtime: 70 secs, T-bit: off, Lan delay: 1, Override interval:
3
 DR priority: 100, Gen ID: 625159467,
 Secondary addresses:
  192.168.30.1
```

show ip pim sparse-mode nexthop

Use this command to see the next hop information as used by PIM-SM.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip pim sparse-mode nexthop`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip pim sparse-mode nexthop
```

Figure 51-10: Example output from the show ip pim sparse-mode nexthop command

```
Flags: N = New, R = RP, S = Source, U = Unreachable
Destination Type Nexthop Nexthop Nexthop      Nexthop    Metric Pref  Refcnt
                 Num     Addr                 Ifindex    Name
-------------------------------------------------------------------------
10.10.0.9   .RS.  1      0.0.0.0  4           0           0        1
```

Table 51-2: Parameters in output of the show ip pim sparse-mode nexthop command

Parameter	Description
`Destination`	The destination address for which PIM-SM requires next hop information.
`Type`	The type of destination, as indicated by the Flags description. N = New, R= RP, S = Source, U = Unreachable.
`Nexthop Num`	The number of next hops to the destination. PIM-SM always uses only 1 next hop.
`Nexthop Addr`	The address of the primary next hop gateway.
`Nexthop IfIndex`	The interface on which the next hop gateway can be reached.
`Nexthop Name`	The name of next hop interface.
`Metric`	The metric of the route towards the destination.
`Preference`	The preference of the route towards destination.
`Refcnt`	Only used for debugging.

show ip pim sparse-mode rp-hash

Use this command to display the Rendezvous Point (RP) to be chosen based on the group selected.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip pim sparse-mode rp-hash <group-addr>`

Parameter	Description
<group-addr>	The group address for which to find the RP, entered in the form A.B.C.D.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip pim sparse-mode rp-hash 224.0.1.3
```

Figure 51-11: Output from the show ip pim sparse-mode rp-hash command

```
RP: 10.10.11.35
Info source: 10.10.11.35, via bootstrap
```

Related Commands show ip pim sparse-mode rp mapping

show ip pim sparse-mode rp mapping

Use this command to show group-to-RP (Rendezvous Point) mappings, and the RP set.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip pim sparse-mode rp mapping`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip pim sparse-mode rp mapping
```

Figure 51-12: Output from the show ip pim sparse-mode rp mapping command

```
PIM Group-to-RP Mappings
Group(s): 224.0.0.0/4
  RP: 10.10.0.9
    Info source: 10.10.0.9, via bootstrap, priority 192
         Uptime: 16:52:39, expires: 00:02:50
```

Related Commands show ip pim sparse-mode rp-hash

undebug all pim sparse-mode

Use this command to disable all PIM-SM debugging.

Syntax `undebug all pim sparse-mode`

Mode Privileged Exec

Example

```
awplus# undebug all pim sparse-mode
```

Related Commands **debug pim sparse-mode**

Chapter 52: PIM-SMv6 Introduction and Configuration

Introduction 52.2
PIM-SMv6 ..... 52.2
Characteristics of PIM-SMv6 ..... 52.3
Roles in PIM-SMv6 ..... 52.4
PIM-SMv6 Terminology 52.5
Operation of PIM-SMv6 ..... 52.6
Data Flow from Source to Receivers for PIM-SMv6 ..... 52.8
PIM-SMv6 Embedded RP, RP and BSR Candidate Configurations 52.10
- Embedded RP Configuration 52.10
- Verify Embedded RP Configuration 52.11
- RP and BSR Candidate Configuration 52.13
- Verify RP and RP Candidate Configuration 52.15

PIM-SMv6 Static RP, DR, BSR Configurations 52.17
- Static Rendezvous Point Configuration 52.18
- Verify Static Rendezvous Point Configuration 52.19
- Dynamic Rendezvous Point Configuration 52.20
- Verify PIM Group-to-RP Mappings 52.21
- Verify RP Details 52.22
- Boot Strap Router Configuration 52.22
- Verify Boot Strap Router Configuration 52.23

Introduction

This chapter provides information about Protocol Independent Multicast - Sparse Mode for IPv6 (PIM-SMv6).

PIM-SMv6

Protocol Independent Multicast - Sparse Mode for IPv6 (PIM-SMv6) provides efficient communication between members of sparsely distributed groups - the type of groups that are most common in wide-area internetworks.

PIM-SMv6 helps geographically dispersed network nodes to conserver bandwidth and reduce traffic by simultaneously delivering a single stream of information to multiple locations. PIM-SMv6 uses the IPv6 multicast model of receiver-initiated membership, supporting both shared and shorted-path trees and uses mechanisms to adapt to changing network conditions. PIM-SMv6 uses a topology gathering approach to populate a multicast routing table with routes.

Note IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Note The IPv6 Multicast addresses shown can be derived from IPv6 unicast prefixes as per RFC 3306.The IPv6 unicast prefix reserved for documentation is 2001:0db8::/32 as per RFC 3849. Using the base /32 prefix the IPv6 multicast prefix for 2001:0db8::/32 is ff3x:20:2001:0db8::/64. Where an RP address is 2001:0db8::1 the embedded RP multicast prefix is ff7x:120:2001:0db8::/96. For ASM (Any-Source Multicast) the IPV6 multicast addresses allocated for documentation purposes are ff0x::0db8:0:0/96 as per RFC 6676. This is a /96 prefix so that it can be used with group IDs as per RFC 3307. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

Note The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

For details of the commands used to configure PIM-SMv6, see **Chapter 53, PIM-SMv6 Commands**. For an overview of multicasting and multicasting commands, see **Chapter 47, Multicast Introduction and Commands**. For an overview of MLD and MLD commands, see **Chapter 56, MLD and MLD Snooping Introduction and Commands**.

Characteristics of PIM-SMv6

PIM Sparse Mode for IPv6 (PIM-SMv6) is designed on the principle that several hosts wishing to participate in a multicast conference does not justify flooding the entire internetwork with periodic multicast traffic. PIM-SMv6 is designed to limit multicast traffic so that only those switches interested in receiving traffic for a particular group receive the traffic.

Switches with directly attached or downstream members are required to join a Sparse Mode distribution tree by transmitting explicit join messages. If a switch does not become part of the predefined distribution tree, it does not receive multicast traffic addressed to the group. The default forwarding action of a sparse mode multicast routing protocol is to block traffic unless it is explicitly requested.

PIM-SMv6 employs the concept of a Rendezvous Point (RP) where receivers “meet” sources. The initiator of each multicast group selects a primary RP and a small ordered set of alternative RPs, known as the RP-list. For each multicast group, there is only a single active RP. Each receiver wishing to join a multicast group contacts its directly attached switch, which in turn joins the multicast distribution tree by sending an explicit join message to the group’s primary RP.

A source uses the RP to announce its presence and to find a path to members that have joined the group. This model requires Sparse Mode switches to maintain some state information (the RP-list) prior to the arrival of data packets.

Roles in PIM-SMv6

A multicast sender does not need to know the addresses of the members of the group in order to send to them, and the members of the group need not know the address of the sender. Group membership can change at any time. When PIM-SMv6 is enabled on the switch, and before the switch can route multicast traffic, it must establish which of the PIM-SMv6 routers in the network are performing some key roles:

- Designated Router.
- Rendezvous Point.
- Bootstrap Router.

Designated Router

There must be one PIM-SMv6 Designated Router (DR) in the subnetwork to which the IPv6 hosts are connected. Any PIM-SMv6 interfaces on the subnetwork elect the DR with the highest DR priority. If there is more than one router with the same priority, or no priority, they choose the interface with the highest IPv6 address number.

The DR performs all the PIM-SMv6 functionality for the subnetwork. If the current DR becomes unavailable, the remaining switches elect a new DR on the interface by DR priority or IPv6 address.

Rendezvous Point

Each multicast group must have a Rendezvous Point (RP). The RP forms the root of the group's distribution tree. The DR for a multicast sender sends multicast packets towards the RP.

DRs with group members connected to them send join messages towards the group's RP. The RP candidate with the lowest priority is elected from all the RP candidates for a group. If the RP becomes unavailable, the remaining RP candidates elect a new RP.

An RP router is configured as the root of the non-source-specific distribution tree for a multicast group. Join messages from receivers for a group are sent towards the RP. Data from senders is sent to the RP so that receivers can discover who the senders are, and receive traffic destined for the group.

Bootstrap Router

Each PIM-SMv6 network must have at least one Bootstrap Router (BSR) candidate, unless all switches in the domain are configured statically with information about all RPs in the domain. Every switch that is a BSR candidate periodically sends a Bootstrap Candidate Advertisement message to advertise that it is available as a BSR candidate.

The BSR candidates in the network elect the switches with the highest preference value to be the BSR. The elected BSR listens to PIM-SMv6 Candidate RP Advertisement messages specifying RP candidates for multicast groups. It maintains a list of RP candidates and sends a bootstrap message every BSM interval, specifying all the multicast groups in the PIM-SMv6 network, and their RP candidates. Each switch uses this information and a standardized hash mechanism to determine the RP for each group. In summary:

- Each multicast group must have at least one RP candidate
- Each PIM-SMv6 domain must have at least one BSR candidate, unless all routers in the domain are configured statically with information about all RPs in the domain
- Each subnetwork must have at least one DR candidate.

PIM-SMv6 Terminology

PIM-SMv6 hello messages

When PIM-SMv6 is enabled on a switch, it sends out a PIM-SMv6 Hello message on all its PIM-SMv6 enabled interfaces, and listens for Hello messages from its PIM-SMv6 neighbors. When a switch receives a Hello message, it records the interface, IPv6 address, priority for becoming a DR, and the timeout for the neighbor's information. The switch sends Hello messages regularly at the Hello Time interval.

Multicast Routing Information Base (MRIB)

The MRIB is a multicast topology table derived from the unicast routing table. In PIM-SMv6, the MRIB decides where to send Join/Prune messages. It also provides routing metrics for destination addresses. These metrics are used when sending and processing Assert messages.

Tree Information Base (TIB)

The TIB is the collection of states at a PIM-SMv6 router storing the state of all multicast distribution trees at that PIM-SMv6 router. It is created by receiving Join/Prune messages, Assert messages, and MLD information from local hosts.

Upstream

Upstream specifies when traffic is going towards the root of the tree. The root of the tree may be either the Source or the RP (Rendezvous Point).

Downstream

Upstream specifies when traffic is going away from the root of the tree. The root of the tree may be either the Source or the RP (Rendezvous Point).

Source-Based Trees

In the Source-Based Trees concept, the forwarding paths are based on the shortest unicast path to the source. If the unicast routing metric is the hop count, then the branches of the multicast Source-Based Trees are the minimum hop. If the routing metric is the delay, then the branches of the multicast Source-Based Trees are the minimum delay.

A corresponding multicast tree directly connects the source to all receivers for every multicast source. All traffic to the members of an associated group passes along the tree made for their source. Source-Based Trees has two entries with a list of outgoing interfaces - the source address and the multicast group.

Shared Trees

Shared Trees, or RP trees (RPT), rely on a central router called the Rendezvous Point (RP) that receives all traffic from the sources, and forwards that traffic to the receivers. Not all hosts may be receivers.

There is a single tree for each multicast group, regardless of the number of sources. Only the routers on the tree know about the group, and information is only sent to interested receivers. With an RP, receivers have a place to join, even if no source exists. The shared tree is unidirectional, and information flows only from the RP to the receivers. If a host other than the RP has to send data on the tree, then the data must first be tunneled to the RP, then multicast to the members. This means that even if a receiver is also a source, it can only use the tree to receive packets from the RP, and to send packets to the RP (unless the source is located between the RP and the receivers).

(*,G)

A 'star G entry' is a PIM-SMv6 or MLD join message that is requesting to join group G (e.g ff0e::1) from any (*) source IPv6 address.

(S,G)

An 'S G entry' is a PIM-SMv6 or MLD join message that is requesting to join group G (e.g ff0e::1) from source IPv6 address (e.g. 2001::1), where S is the source IPv6 address that is generating the multicast data for G. Note that PIM-SMv6 supports (S,G) entries, but MLDv1 does not support (S,G) entries.

BSM

Boot Strap Messages, as issued by the BSR (Boot Strap Router), which is an elected router that distributes information about the RP (Rendezvous Point), where an RP is a router in a multicast network domain that acts as a shared route for a multicast shared tree.

MLD

Multicast Listener Discovery. There are two versions: MLDv1 and MLDv2. MLDv1 is used by an IPv6 router to discover the presence of multicast listeners. MLDv2 provides additional features such as the ability to specify a source IPv6 address when sending a join.

Operation of PIM-SMv6

Once roles are established, multicast routing follows specific phases:

1. **Rendezvous Point Tree**
2. **Register Stop**
3. **Shortest Path Tree**

While multicast routing always begins with phase 1, the Designated Router (DR) for a receiver determines whether and when to move on to phases 2 and 3, depending on the amount of traffic from the source.

Rendezvous Point Tree

Phase 1 establishes and uses a shared tree rooted at the Rendezvous Point (RP) to forward all multicast data to group members.

When an IPv6 host sends an MLD join message to the local PIM DR, which is not the RP for the group, the DR sends a PIM-SMv6 join message towards the RP for the group ("upstream"). The DR determines which switch is the RP for the group from the most recent bootstrap message. Every switch the join message passes through records that there is a group member on the incoming interface.

Eventually, the join message reaches either the RP, or another switch that already knows that it has a group member downstream. If the group has many members, the join messages converge on the RP to form a Rendezvous Point Tree (RPT). This is called a shared tree because multicast data that is sent to the group by any sender shares the tree.

The multicast receiver's DR sends join messages periodically according to the upstream join timer as long as the IPv6 host is a member of the group. When the last receiver on a subnetwork leaves the group, the join messages stop, and their entries timeout on routers that are closer to the RP.

The sender's DR encapsulates the multicast data in a unicast packet in a process called **registering**, and sends these register packets to the group's RP. When the RP receives the data, it decapsulates them, and forwards them onto the shared tree.

Register Stop

Phase 2 improves efficiency and performance by using register stop. In this phase the RP joins the shortest path tree between the source and receiver. This allows the original (unencapsulated) packets to be forwarded from the sender, instead of encapsulated packets. It also allows shorter paths to receivers that are close to the sender, making it more efficient in some circumstances.

When the RP for a group receives the first encapsulated data packet from a source, it joins the shortest path tree towards the sender. Once data is able to flow along the shortest path from the sender to the RP, packets do not need to be registered. The RP sends a register stop message in reply to the next encapsulated message. When the sender's DR receives the register stop message, it stops registering.

The DR sends a null register message to the RP to find whether the RP still does not need to receive registered packets. If it receives another register stop message, the DR continues to forward only the native data packets. If the DR does not receive another register stop message within the register probe time, it resumes registering the data packets and sending them to the RP.

When the RP starts receiving native data packets from the source, it starts to discard the encapsulated packets, and starts forwarding native packets on the shared tree to all the group members. If the path from the source to the RP intersects the shared RP tree for the group, then the packets also take a short-cut onto the shared tree for delivery to the group members down its branches.

Shortest Path Tree

This phase further optimizes routing by using Shortest Path Trees (SPT). In phase 3 the receiver joins the shortest path tree between the source and receiver. This allows a multicast group member to receive multicast data by the shortest path from the sender, instead of from the shared RP tree. When the receiver's DR receives multicast data from a particular sender, it sends a join message towards the sender. When this message reaches the sender's DR, the DR starts forwarding the multicast data directly towards the receiver. As several receivers all initiate shortest paths to the sender, these paths converge, creating a SPT.

When the multicast packets start arriving from the SPT at the receiver's DR or an upstream router common to the SPT and the RPT, it starts discarding the packets from the RPT, and sends a prune message towards the RP. The prune message travels up the RPT until it reaches the RP or a switch that still needs to forward multicast packets from this sender to other receivers. Every time a switch receives a prune message, it waits a short time so that other switches on the LAN have the opportunity to override the prune message.

Multi-Access LANs

If the PIM-SMv6 network includes multi-access LAN links for transit, as well as point-to-point links, then a mechanism is needed to prevent multiple trees forwarding the same data to the same group member. Two or more switches on a LAN may have different information about how to reach the RP or the multicast sender. They could each send a join message to two different switches closer to the RP for an RPT or the sender for an SPT. This could potentially cause two copies of all the multicast traffic towards the receiver.

When PIM -SMv6 switches notice duplicate data packets on the LAN, they elect a single switch to forward the data packets, by each sending PIM-SMv6 Assert messages. If one of the upstream switches is on an SPT and the other is on an RPT, the switch on the SPT has the shortest path to the sender, and wins the Assert election. If both switches are on RPTs the switch with the shortest path to the RP (the lowest sum of metrics to the RP) wins the Assert. If both switches are on an SPT, then the switch with the shortest path to the sender (the lowest sum of metrics to the sender's DR) wins the Assert.

The switch that won the Assert election forwards these data packets, and acts as the local Designated Router for any MLD members on the LAN. The downstream switches on the LAN also receive the Assert messages, and send all their join messages to the Assert winner. The result of an Assert election will timeout after the Assert Time. As long as the situation causing the duplication remains unchanged, the Assert winner sends an Assert message at a the Assert time interval, before the previous Assert messages time out. When the last downstream switch leaves the SPT, the Assert winner sends an Assert Cancel message saying that it is about to stop forwarding data on the SPT. Any RPT downstream switches then switch back to the RP tree.

Packet Types

See the below list of PIM-SMv6 IPv6 packet types:

- **Register Packets** - unicast encapsulated multicast packets sent to the RP.
- **Hello Messages** - periodic messages sent by PIM-SMv6 routers.
- **Register-Stop Messages** - send to stop the flow of register encapsulated packets to the RP from a source.
- **Join/Prune Messages** - sent when a client joins/leaves a group to maintain PIM-SMv6.
- **Assert Messages** - sent to resolve route conflicts.

RFCs

PIM-SMv6 implements the service model from **RFC 1112** to route packets from sources to receivers with either the source or the receiver knowing of each other. This is done in three phases, which may be done at the same time. Clients join or leave multicast groups using MLD protocols. See **Chapter 56, MLD and MLD Snooping Introduction and Commands**

The operation of PIM-SMv4 and PIM-SMv6 protocols is the same. Only the addresses change from IPv4 to IPv6. See **RFC 4601** for further details about the operation of PIM-SM.

Data Flow from Source to Receivers for PIM-SMv6

See the details below for the data flow from Source to Receivers to establish the phases of the Rendezvous Point Tree, Register Stop and Shortest Path Tree as described previously:

1. **Sending out Hello Messages**
2. **Electing a Designated Router**
3. **Determining the RP**
4. **Joining the Shared Tree**
5. **Registering with the RP**
6. **Sending Register-Stop Messages**
7. **Pruning the Interface**
8. **Forwarding Multicast Packets**

Sending out Hello Messages

PIM-SMv6 router periodically send Hello messages to discover neighboring PIM-SMv6 routers. PIM-SMv6 routers do not send any acknowledgment that a Hello message was received. A holdtime value determines the length of time for which the information is valid. In PIM-SMv6, a downstream receiver must join a group before traffic is forwarded on the interface.

Electing a Designated Router

In a multi-access network with multiple PIM-SMv6 routers connected, one of the PIM-SMv6 routers is selected to act as a Designated Router (DR) for a given period. The DR is responsible for sending Join/Prune messages to the RP for local members.

Determining the RP

PIM-SMv6 uses a Boot Strap Router (BSR) to originate Boot Strap Messages (BSM, and to disseminate RP information.

The BSM messages are multicast to the group on each link. If the BSR is not apparent, then the router flood the domain with advertisements. The router with the highest priority (if priorities are the same, the highest IPv6 address is applied) is selected to be the RP. PIM-SMv6 routers receive and store BSM messages originated by the BSR.

When a DR gets a membership indication from MLD for (or a data packet from) a directly connected host, for a a group for which it has no entry, the DR maps the group address to one of the candidate RPs that can service that group. The DR then sends a Join/Prune message towards that RP. In a small domain, the RP can also be configured statically.

Joining the Shared Tree

To join a multicast group, a host sends an MLD message to its upstream PIM-SMv6 router, after which the PIM-SMv6 router can accept multicast traffic for that group. The PIM-SMv6 router sends a Join message to its upstream PIM-SMv6 neighbor in the direction of the RP.

When a PIM-SMv6 router receive a Join message from a downstream PIM-SMv6 router, it checks to see if a state exists for the group in its multicast routing table.

If a state already exists, the Join message has reached the shared tree, and the interface from which the message was received is entered in the Outgoing Interface list.

If no state exists, an entry is created, the interface is entered in the Outgoing Interface lists, and the Join message is again sent towards the RP.

Registering with the RP

A DR can begin receiving traffic from a source without having a Source or Group state for that source. In this case, the DR has no information on how to get multicast traffic to the RP through a tree. When the source DR receives the initial multicast packet, it encapsulates it in a Register message, and unicasts it to the RP for that group. The RP de-encapsulates each Register message, and forwards the extracted data packet to downstream members on the RPT.

Once the path is established from the source to the RP, the DR begins sending traffic to the RP as standard IP multicast packets, as well as encapsulated within Register messages. The RP temporarily receives packets twice. The RP detects the normal multicast packets, it sends a Register-Stop message to the source DR, meaning it should stop sending register packets.

Sending Register-Stop Messages

When the RP begins receiving traffic from the source, both as Register messages and as unencapsulated IPv6 packets, it sends a Register-Stop message to the DR. This notifies the DR that the traffic is now being received as standard IPv6 multicast packets on the SPT. When the DR receives this message, it stops encapsulating traffic in Register messages.

Pruning the Interface

PIM-SMv6 Routers attached to receivers send Prune messages to the RP to disassociate the source from the RP. When an RP receives a Prune message, it no longer forwards traffic from the source indicated in the Prune message.

If all members of a multicast group are pruned, then the MLD state of the DR is deleted, and the interface is removed from the Source and Group lists of the group.

Forwarding Multicast Packets

PIM-SMv6 routers forward multicast traffic onto all interfaces that lead to receivers that have explicitly joined a multicast group. Messages are sent to a group address in the local subnetwork, and have a Time to Live (TTL) of 1. The PIM-SMv6 router performs an RPF check, and forwards the packet.

If a downstream PIM-SMv6 router has sent a join to this PIM-SMv6 router, or is a member of this group, then traffic that arrives on the correct interface is sent to all outgoing interfaces that lead to downstream receivers.

PIM-SMv6 Embedded RP, RP and BSR Candidate Configurations

This section provides two PIM-SMv6 configuration examples:

- **Embedded RP Configuration**
- **RP and BSR Candidate Configuration**

For details on the commands used in the following examples, refer to **Chapter 53, PIM-SMv6 Commands**.

Embedded RP Configuration

RFC 3956 describes a multicast address allocation policy, in which the address of the Rendezvous Point (RP) is encoded in the IPv6 multicast group address, and specifies a PIM-SMv6 group-to-RP mapping to use the encoding, leveraging and extending unicast-prefix-based addressing.

Embedded RP Multicast Group Address Format

RFC 3956 specifies a modification to the unicast-prefix-based address format by specifying the second high-order bit (R-bit) as follows:

Table 52-1: RFC3956 modification to the unicast-prefix-based IPv6 address format

11111111	flgs	scop	rsvd	RIID	plen	network prefix	group ID
8 bits	4 bits	4 bits	4 bits	4 bits	8 bits	64 bits	32 bits

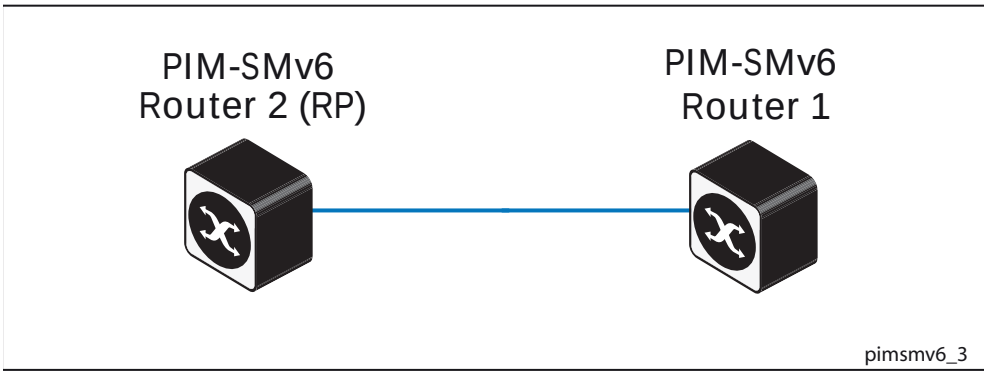
Note: **flgs** is a set of four flags - **0** - **R** - **P** - **T**

flgs the highest-order bit is 0, flag R is 1. This indicates a multicast address that embeds the address on the RP. In this case, the **P** flag must be set to 1 and the **T** flag must be set to 1. In effect, this implies the prefix **ff70::/12**. This means that the last 4 bits of the previously reserved field are interpreted as the embedded RP interface ID.

RP Address in Embedded RP Multicast Address

The address of the RP can only be embedded in unicast-prefix-based Any Source Multicast (ASM) addresses. To identify whether it is an embedded RP multicast address, and to be processed any further, an IPv6 address must satisfy all of the following criteria:

- It must be an IPv6 multicast address with **flgs** set to **0111** to be of the prefix **ff70::/12,** or with **flgs** set to **1111** to be of the prefix **fff0::/12**
- **plen** must not be set to 0 and **plen** must not be set to greater than 64.



Enable embedded RP

Enter the following commands to enable embedded-RP to group mapping (ipv6 pim rp embedded) then configure an access-list to permit a multicast group (ipv6 access-list) and use this access-list to limit valid groups with a configured static RP (ipv6 pim rp-address):

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim rp embedded
awplus(config)# ipv6 access-list embedrp1 permit
ff70:2:2001:0db8:12::2::/96
awplus(config)# ipv6 pim rp-address ff70:2001:0db8:12::2/12
embedrp1
awplus(config)# exit
```

Disable embedded RP

Enter the following commands to disable embedded-RP to group mapping (no ipv6 pim rp embedded) after enabling embedded RP (as shown in the previous example above):

```
awplus# configure terminal
awplus(config)# no ipv6 pim rp embedded
awplus(config)# exit
```

PIM-SMv6 commands used

ipv6 pim rp-address
ipv6 pim rp embedded

Verify Embedded RP Configuration

Use the following commands to verify the embedded-RP configuration. Note that the group-to-RP mapping for embedded-RP addresses is created when the group is first seen at a PIM-SMv6 router. This can be due to the MLD local receiver report, Join/Prune and Register message processing.

Verify RP-mapping in RP:

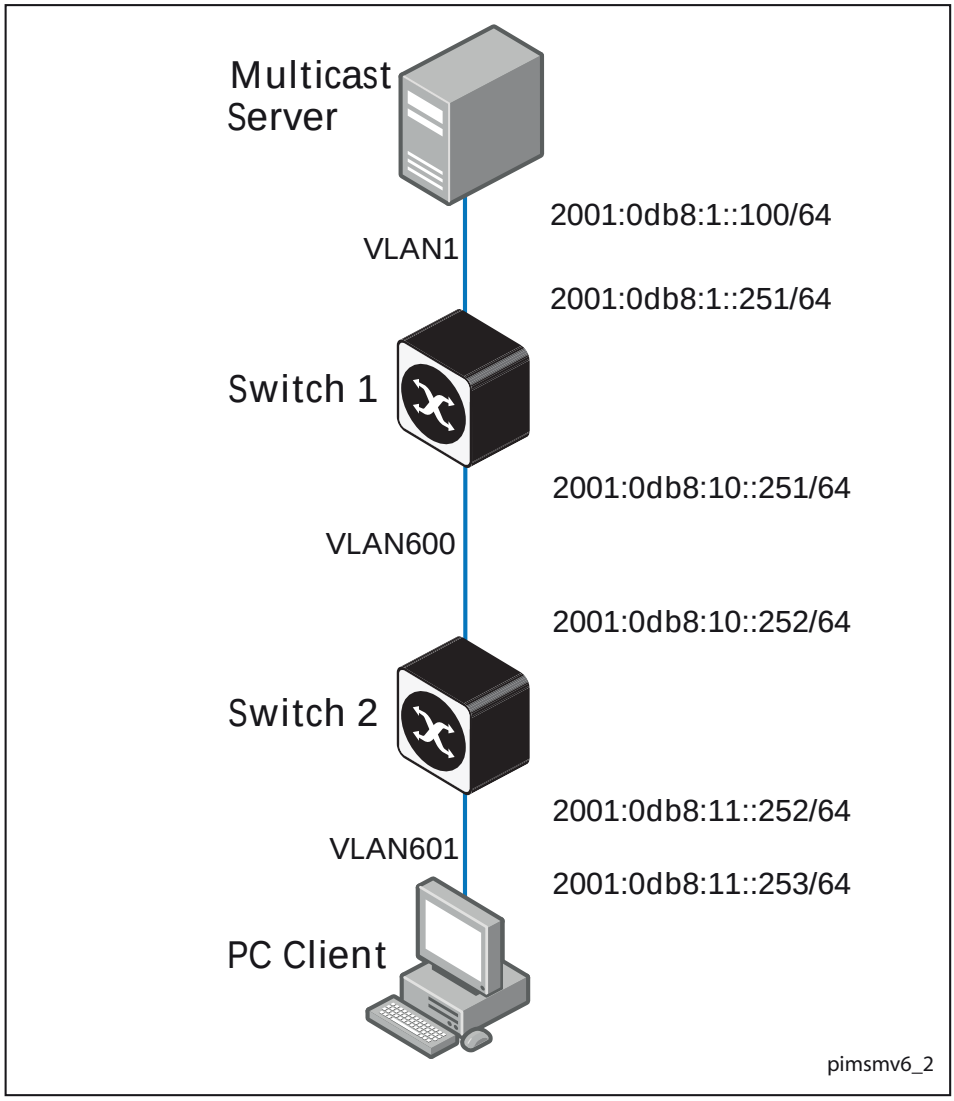
```
awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): ff7e:240:3ffe:172:31:12::/96, Static
    RP: 3ffe:172:31:12::2
         Uptime: 00:04:12
Embedded RP Groups:
Group(s): ff7e:240:3ffe:172:31:12::/96
    RP: 3ffe:172:31:12::2, Uptime: 00:00:33
```

Verify RP-mapping in non-RP:

```
mv66#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Embedded RP Groups:
Group(s): ff7e:240:3ffe:172:31:12::/96
    RP: 3ffe:172:31:12::2, Uptime: 00:00:27
```

RP and BSR Candidate Configuration

Apply the configurations for **Switch 1** and **Switch 2** shown following the topology to configure **Switch 1** and **Switch 2** as RP and BSR candidates.



Note the **Multicast Server** serves multiple streams on addresses **ff0e:1:1::3** and **ff0e:3:1::4**. Also note **Switch 1** and **Switch 2** are configured as BSR Candidates with **Switch 2** elected.

Switch 1 Enter this configuration to configure **Switch 1** as an RP candidate and a BSR candidate:

```
!
no ip multicast-routing
!
ipv6 multicast-routing
!
vlan database
 vlan 600 state enable
!
ipv6 pim bsr-candidate vlan600
ipv6 pim rp-candidate vlan600
!
!
interface port1.0.17
 switchport
 switchport mode access
 switchport access vlan 1
!
interface port1.0.24
 switchport
 switchport mode access
 switchport access vlan 600
!
interface vlan1
 ip address 192.168.1.101/24
 ipv6 address 2001:0db8:1::251/64
 ipv6 enable
 ipv6 router rip
 ipv6 pim sparse-mode passive
!
interface vlan600
 ipv6 address 2001:0db8:10::251/64
 ipv6 enable
 ipv6 router rip
 ipv6 pim sparse-mode
!
ipv6 forwarding
!
```

Switch 2 Enter this configuration to configure **Switch 2** as an RP candidate and a BSR candidate:

```
!
ipv6 access-list standard testacl permit ff0e:1::3/128
!
no ip multicast-routing
!
ipv6 multicast-routing
!
vlan database
 vlan 600-601 state enable
!
ipv6 pim bsr-candidate vlan600
ipv6 pim rp-candidate vlan601 group-list testacl
!
interface port1.0.11
 switchport
 switchport mode access
 switchport access vlan 600
!
interface port1.0.13-1.0.14
 switchport
 switchport mode access
 switchport access vlan 603
!
interface port1.0.15-1.0.19
 switchport
 switchport mode access
!
interface port1.0.20
 switchport
 switchport mode access
 switchport access vlan 601
!
interface port1.0.21-1.0.24
 switchport
 switchport mode access
!
interface vlan600
 ipv6 address 2001:0db8:10::252/64
 ipv6 enable
 ipv6 router rip
 ipv6 pim sparse-mode
!
interface vlan601
 ipv6 address 2001:0db8:11::252/64
 ipv6 enable
 ipv6 mld
 ipv6 router rip
 ipv6 pim sparse-mode
!
ipv6 forwarding
!
```

Verify RP and RP Candidate Configuration

Use the following commands to verify the RP candidate configuration for **Switch 1** and **Switch 2**. Note both **Switch 1** and **Switch 2** are BSR candidates and **Switch 2** is elected.

Also note **Switch 1** is a candidate RP for all multicast groups, while **Switch 2** is a candidate for only the multicast group **ff0e:1::3/128**. Therefore, **Switch 1** and **Switch 2** overlap and **Switch 1** has become the RP for **ff0e:1::4** while Switch 2 is now the RP for **ff0e:1::3**.

Verify Switch 1 configuration:

```
awplus#show ipv6 pim sparse-mode bsr-router
PIM6v2 Bootstrap information
  BSR address: 2001:0db8:10::252 (?)
  Uptime:      01:09:46, BSR Priority: 64, Hash mask length: 126
  Expires:     00:01:44
  Role: Candidate BSR
  State: Candidate BSR

  Candidate RP: 2001:0db8:10::251(vlan600)
    Advertisement interval 60 seconds
    Next C-RP advertisement in 00:00:21

awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): ff00::/8
  RP: 2001:0db8:10::251
    Info source: 2001:0db8:10::252, via bootstrap, priority 192
         Uptime: 01:24:33, expires: 00:02:24
Group(s): ff0e:1::3/128
  RP: 2001:11::252
    Info source: 2001:0db8:10::252, via bootstrap, priority 192
         Uptime: 00:34:06, expires: 00:02:24
Embedded RP Groups:

awplus#show ipv6 pim sparse-mode rp-hash ff0e:1::3
    RP: 2001:0db8:11::252
    Info source: 2001:0db8:10::252, via bootstrap
```

Verify Switch 2 configuration (Switch 2 is the elected BSR):

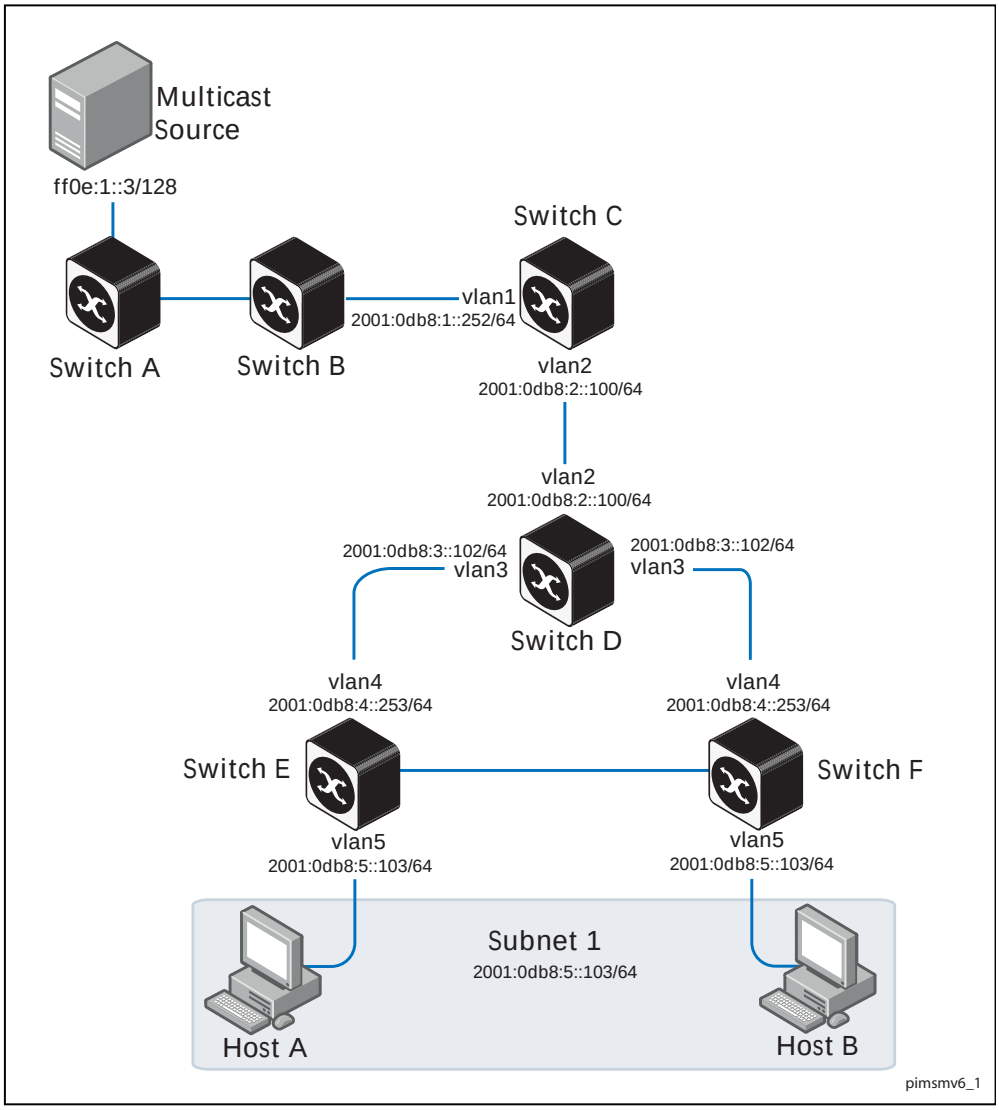
```
awplus#show ipv6 pim sparse-mode bsr-router
PIM6v2 Bootstrap information
This system is the Bootstrap Router (BSR)
  BSR address: 2001:0db8:10::252 (?)
  Uptime:      01:08:31, BSR Priority: 64, Hash mask length: 126
  Next bootstrap message in 00:00:49
  Role: Candidate BSR
  State: Elected BSR

  Candidate RP: 2001:0db8:11::252(vlan601)
    Advertisement interval 60 seconds
    Next C-RP advertisement in 00:00:44

awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
This system is the Bootstrap Router (v2)
Group(s): ff00::/8
  RP: 2001:0db8:10::251
    Info source: 2001:0db8:10::251, via bootstrap, priority 192
         Uptime: 01:17:06, expires: 00:01:46
Group(s): ff0e:1::3/128
  RP: 2001:11::252
    Info source: 2001:0db8:10::252, via bootstrap, priority 192
         Uptime: 00:37:41, expires: 00:01:54
Embedded RP Groups:

awplus#show ipv6 pim sparse-mode rp-hash ff0e:1::4
    RP: 2001:0db8:10::251
    Info source: 2001:0db8:10::252, via bootstrap
```

PIM-SMv6 Static RP, DR, BSR Configurations

This section provides three PIM-SMv6 configuration examples:

- **Static Rendezvous Point Configuration**
- **Dynamic Rendezvous Point Configuration**
- **Boot Strap Router Configuration**

Both Rendezvous Point (RP) configuration examples refer to the network topology in the following graphic and use Allied Telesis managed Layer 3 Switches as the PIM routers. For details on the commands used in the following examples, refer to **Chapter 53, PIM-SMv6 Commands**.



Static Rendezvous Point Configuration

In this example using the above network topology, Switch C is the Rendezvous Point (RP) and all switches are statically configured with RP information. Host A and Host B join group `ff0e:1::3/128` for all the sources. They send the MLD membership report to Subnet 1. Two switches are attached to Subnet 1, Switch E and Switch F. Both of these switches have default Designated Router (DR) priority on `vlan1`. Because Switch E has a higher IP address on `vlan1`, Switch E becomes the DR and is responsible for sending Join messages to the RP (Switch C).

While configuring the RP, ensure that:

- Every switch includes the **ipv6 pim rp-address 2001:0db8:1::100/64** statement, even if it does not have any source or group member attached to it.
- There is only one RP address for the whole multicast group.
- All interfaces running PIM-SMv6 must have sparse-mode enabled. In the configuration sample output below, both **vlan1** and **vlan2** are pim sparse-mode enabled.

See the following configuration output for **Switch D**:

```
hostname Switch D
!
interface vlan2
 ipv6 enable
 ipv6 pim sparse-mode
 ipv6 address 2001:0db8:2::100/64
!
interface vlan3
 ipv6 enable
 ipv6 pim sparse-mode
 ipv6 address 2001:0db8:3::102/64
!
interface lo
!
!
ipv6 multicast-routing
ipv6 pim rp-address 2001:0db8:1::100/64
!
ipv6 forwarding
!
```

Configure all the switches with the same **ipv6 pim rp-address 2001:0db8:1::100/64** statement as shown above.

Verify Static Rendezvous Point Configuration

Use the following commands to verify the RP configuration, interface details, and the multicast routing table.

RP details

For **Switch D**, the **show ipv6 pim sparse-mode rp mapping** command shows that `2001:0db8:1::100/64` is the RP for all multicast groups `ff0e:1::3/128`, and is statically configured. All other switches will have a similar output.

```
awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): ff0e:1::3/128, Static
    RP: 2001:0db8:1::100/64
         Uptime: 5d02h15m
```

For **Switch D**, the **show ipv6 pim sparse-mode rp-hash** command displays the selected RP for the specified group, in this example `ff0e:1::3/128`.

```
awplus#show ipv6 pim sparse-mode rp-hash ff0e:1::3/128
    RP: 2001:0db8:1::100/64
```

Interface details

For **Switch E**, the **show ipv6 pim sparse-mode interface** command displays the interface details and shows that Switch E is the DR on Subnet 1.

```
awplus#show ipv6 pim sparse-mode interface
Total configured interfaces: 16   Maximum allowed: 31
Total active interfaces:     12

Address             Interface VIFindex Ver/   Nbr      DR       DR
                                       Mode   Count    Prior
2001:0db8:1::251    vlan2      0       v2/S   1        1        2001:0db8:1::100
```

IP multicast routing table

Note that the multicast routing table displayed for an RP switch is different to that displayed for other switches. For **Switch C**, because this switch is the RP and the root of this multicast tree, the **show ipv6 pim sparse-mode mroute** command shows **RPF nbr** (next-hop to reach RP) as **0.0.0.0** and **RPF idx** (incoming interface for this (*, G) state) as **None**.

```
awplus#show ipv6 pim sparse-mode mroute
IP Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
(*, ff0e:1::3/128)
RP: 2001:0db8:1::100/64
RPF nbr: 0.0.0.0
RPF idx: None
Upstream State: JOINED
 Local     ..............................
 Joined    j..............................
 Asserted  ..............................
 Outgoing  o..............................
```

For **Switch E**, the **show ipv6 pim sparse-mode mroute** command displays the IP multicast routing table.

```
awplus#show ipv6 pim sparse-mode mroute
IP Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
(*, ff0e:1::3/128)
RP: 2001:0db8:1::100/64
RPF nbr: 2001:0db8:1::100/64
RPF idx: port1.0.2
Upstream State: JOINED
 Local      ...............................
 Joined     j...............................
 Asserted   ...............................
 Outgoing   o...............................
```

On Switch E, `port1.0.2` is the incoming interface of the (*, G) entry, and `port1.0.1` is on the outgoing interface list of the (*, G) entry. This means that there is a group member through `port1.0.1`, and RP is reachable through `port1.0.2`.

Dynamic Rendezvous Point Configuration

A static RP configuration works for a small, stable PIM domain. However, it is not practical for a large and not so stable internetwork. In such a network, if the RP fails, the network administrator may have to change the static configurations on all PIM switches. An additional reason for choosing dynamic configuration high routing traffic leading to a change in the RP.

The Bootstrap Router (BSR) mechanism is used to dynamically maintain the RP information. To configure the RP dynamically in the above network topology, Switch C on `port1.0.1` and Switch D on `vlan1` are configured as RP candidates using the **ipv6 pim rp-candidate** command. Switch D on `vlan1` is also configured as the BSR candidate. Since no other device has been configured as a BSR candidate, Switch D becomes the BSR router and is responsible for sending group-to-RP mapping information to all other PIM switches in this PIM domain.

The following output displays the complete configuration at **Switch C**.

```
awplus#show run
!
interface vlan1
 ipv6 enable
 ipv6 pim sparse-mode
 ipv6 address 2001:0db8:1::252/64
!
interface vlan2
 ipv6 enable
 ipv6 pim sparse-mode
 ipv6 address 2001:0db8:2::100/64
!
interface lo
!
ipv6 multicast-routing
ipv6 pim rp-candidate vlan1
```

The following output displays the complete configuration at **Switch D**.

```
awplus#show run
!
interface vlan2
 ipv6 enable
 ipv6 pim sparse-mode
 ipv6 address 2001:0db8:2::100/64
!
interface vlan3
 ipv6 enable
 ipv6 pim sparse-mode
 ipv6 address 2001:0db8:3::102/64
!
interface lo
!
ipv6 multicast-routing
ipv6 pim bsr-candidate vlan1
ipv6 pim rp-candidate vlan1 priority 2
!
```

The highest priority switch is chosen as the RP. If two or more switches have the same priority, a hash function in the BSR mechanism is used to choose the RP to make sure that all devices in the PIM domain have the same RP for the same multicast group.

Use the `<interface> [priority <priority>]` parameters of the **ipv6 pim rp-candidate** command to change the default priority of any RP candidate.

Verify PIM Group-to-RP Mappings

The **show ipv6 pim sparse-mode rp mapping** command displays the group-to-RP mapping details. The output shows information about RP candidates. There are two RP candidates for the group range `ff0e:1::3/128`. RP candidate `2001:1::100/64` has a default priority of `192`, whereas RP candidate `2001:1::251/64` has been configured to have a priority of `2`. Since RP candidate `2001:1::251/64` has a higher priority, it is selected as the RP for the multicast group `ff0e:1::3/128`.

See the following configuration output for **Switch D**.

```
awplus#show ipv6 pim sparse-mode rp mapping
This system is the Bootstrap Router (v2)
Group(s): ff0e:1::3/128
  RP: 2001:0db8:1::100/64
    Info source: 2001:0db8:1::251/64, via bootstrap, priority 192
         Uptime: 00:00:13, expires: 00:02:29
```

Verify RP Details

The **show ipv6 pim sparse-mode rp-hash** command displays information about the RP router for a particular group. See the following configuration output for **Switch D**. This output shows that `2001:0db8:1::251/64` has been chosen as the RP for the multicast group `ff0e:1::3/128`.

```
awplus#show ipv6 pim sparse-mode rp-hash ff0e:1::3/128
Group(s): ff0e:1::3/128
    RP: 2001:0db8:1::251/64
    Info source: 2001:0db8:1::251/64, via bootstrap
```

After RP information reaches all PIM switches in the domain, various state machines maintain all routing states as the result of Join/Prune messages from members of the multicast group.

Boot Strap Router Configuration

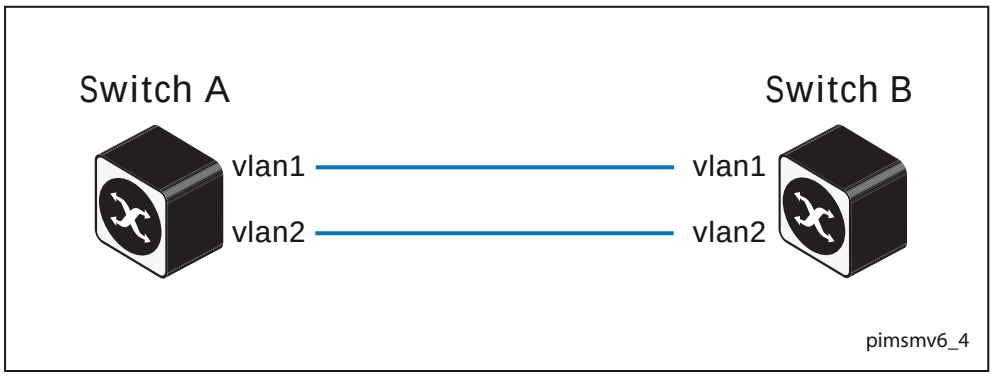
Every PIM multicast group needs to be associated with the IP address of a Rendezvous Point (RP). This address is used as the root of a group-specific distribution tree, whose branches extend to all nodes in the domain that want to receive traffic sent to the group. For all senders to reach all receivers, all devices in the domain use the same mappings of group addresses to RP addresses. In order to determine the RP for a multicast group, a PIM device maintains a collection of group-to-RP mappings, called the RP-Set.

The Boot Strap Router (BSR) mechanism for the class of multicast routing protocols in the PIM domain uses the concept of an RP as a means for receivers to discover the sources that send to a particular multicast group. The BSR mechanism is one way that a multicast router can learn the set of group-to-RP mappings required in order to function.

Some of the PIM devices within a PIM domain are configured as RP candidates. A subset of the RP candidates will eventually be used as the actual RPs for the domain. An RP configured with a lower value in the priority field has higher a priority.

Some of the PIM devices in the domain are configured to be BSR candidates. One of these BSR candidates is elected to be the BSR for the domain, and all PIM devices in the domain learn the result of this election through Bootstrap messages (BSM). The BSR candidate with highest value in the priority field is the elected BSR.

The RP candidates then report their candidacy to the elected BSR, which chooses a subset of the RP candidates, and distributes corresponding group-to-RP mappings to all the devices in the domain through Bootstrap messages.



Switch A Enter the following commands to configure `vlan1` on Switch A as the BSR candidate. The default priority is 64.

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim bsr-candidate vlan1
awplus(config)# exit
```

Switch B Enter the following commands to configure `vlan1` on Switch B as the BSR candidate with a hash mask length of 10 and a priority of 25 and to configure `vlan1` as the RP candidate with a priority of 0.

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim bsr-candidate vlan1 10 25
awplus(config)# ipv6 pim rp-candidate vlan1 priority 0
awplus(config)# exit
```

Verify Boot Strap Router Configuration

Use the **show ipv6 pim sparse-mode bsr-router** command to verify the BSR candidate state on **Switch A**.

```
awplus#show ipv6 pim sparse-mode bsr-router
PIMv2 Bootstrap information
This system is the Bootstrap Router (BSR)
  BSR address: 2001:0db8:1::251/64
  Uptime:      00:37:12, BSR Priority: 64, Hash mask length: 10
  Expires:     00:01:32
  Role: Candidate BSR
  State: Elected BSR
```

Use the **show ipv6 pim sparse-mode bsr-router** command to verify the BSR candidate state on **Switch B**. The initial state of the BSR candidate is pending before transitioning to BSR candidate.

```
awplus#show ipv6 pim sparse-mode bsr-router
PIMv2 Bootstrap information
  BSR address: 2001:0db8:1::251/64
  Uptime:      00:02:39, BSR Priority: 64, Hash mask length: 10
  Expires:     00:00:03
  Role: Candidate BSR
  State: Pending BSR

awplus#show ipv6 pim sparse-mode bsr-router
PIMv2 Bootstrap information
 BSR address: 2001:0db8:1::251/64
 Uptime:      00:40:20, BSR Priority: 64, Hash mask length: 10
 Expires:     00:02:07
 Role: Candidate BSR
 State: Candidate BSR
```

Use the **show ipv6 pim sparse-mode rp mapping** command to verify RP-set information on **Switch A**.

```
awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
This system is the Bootstrap Router (v2)
Group(s): ff0e:1::3/128
 RP: 2001:db8:1::251/64
 Info source: 2001:db8:1::251/64, via bootstrap, priority 0
 Uptime: 00:00:30, expires: 00:02:04
```

Use the **show ipv6 pim sparse-mode rp mapping** command to verify RP-set information on **Switch B**.

```
awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): ff0e:1::3/128
 RP: 2001:db8:1::251/64
 Info source: 2001:db8:1::251/64, via bootstrap, priority 0
 Uptime: 00:00:12, expires: 00:02:18
```

Chapter 53: PIM-SMv6 Commands

Command List 53.2
clear ipv6 mroute pim 53.3
clear ipv6 mroute pim sparse-mode 53.3
clear ipv6 pim sparse-mode bsr rp-set * 53.4
debug ipv6 pim sparse-mode 53.5
debug ipv6 pim sparse-mode packet 53.7
debug ipv6 pim sparse-mode timer 53.8
ipv6 pim accept-register 53.10
ipv6 pim anycast-rp 53.11
ipv6 pim bsr-border 53.12
ipv6 pim bsr-candidate 53.13
ipv6 pim cisco-register-checksum 53.14
ipv6 pim cisco-register-checksum group-list 53.15
ipv6 pim crp-cisco-prefix 53.16
ipv6 pim dr-priority 53.17
ipv6 pim exclude-genid 53.18
ipv6 pim ext-srcs-directly-connected 53.19
ipv6 pim hello-holdtime 53.20
ipv6 pim hello-interval 53.21
ipv6 pim ignore-rp-set-priority 53.22
ipv6 pim jp-timer 53.23
ipv6 pim neighbor-filter 53.24
ipv6 pim register-rate-limit 53.25
ipv6 pim register-rp-reachability 53.26
ipv6 pim register-source 53.27
ipv6 pim register-suppression 53.28
ipv6 pim rp-address 53.29
ipv6 pim rp-candidate 53.31
ipv6 pim rp embedded 53.32
ipv6 pim rp-register-kat 53.33
ipv6 pim sparse-mode 53.34
ipv6 pim sparse-mode passive 53.35
ipv6 pim spt-threshold 53.36
ipv6 pim spt-threshold group-list 53.37
ipv6 pim unicast-bsm 53.38
show debugging ipv6 pim sparse-mode 53.39
show ipv6 pim sparse-mode bsr-router 53.40
show ipv6 pim sparse-mode interface 53.41
show ipv6 pim sparse-mode interface detail 53.42
show ipv6 pim sparse-mode local-members 53.43
show ipv6 pim sparse-mode mroute 53.44
show ipv6 pim sparse-mode mroute detail 53.46
show ipv6 pim sparse-mode neighbor 53.48
show ipv6 pim sparse-mode nexthop 53.49
show ipv6 pim sparse-mode rp-hash 53.50
show ipv6 pim sparse-mode rp mapping 53.51
show ipv6 pim sparse-mode rp nexthop 53.52
undebug all ipv6 pim sparse-mode 53.53
undebug ipv6 pim sparse-mode 53.54

Command List

This chapter provides an alphabetical reference of PIM-SMv6 commands. For IPv6 Multicast commands, see **Chapter 47, Multicast Introduction and Commands**. For an overview of PIM-SMv6, see **Chapter 52, PIM-SMv6 Introduction and Configuration**.

Note

IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous IPv6 static multicast routes.

Note

The IPv6 Multicast addresses shown can be derived from IPv6 unicast prefixes as per RFC 3306.The IPv6 unicast prefix reserved for documentation is 2001:0db8::/32 as per RFC 3849. Using the base /32 prefix the IPv6 multicast prefix for 2001:0db8::/32 is ff3x:20:2001:0db8::/64. Where an RP address is 2001:0db8::1 the embedded RP multicast prefix is ff7x:120:2001:0db8::/96. For ASM (Any-Source Multicast) the IPV6 multicast addresses allocated for documentation purposes are ff0x::0db8:0:0/96 as per RFC 6676. This is a /96 prefix so that it can be used with group IDs as per RFC 3307. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

Note

The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

clear ipv6 mroute pim

Use this command to clear all Multicast Forwarding Cache (MFC) entries in PIM-SMv6.

Note Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Syntax

```
clear ipv6 mroute [*] pim sparse-mode
```

Parameter	Description
*	Clears all PIM-SMv6 multicast routes. Using this command without this optional operator only deletes the multicast router table entries.

Mode Privileged Exec

Example

```
awplus# clear ipv6 mroute pim sparse-mode

awplus# clear ipv6 mroute * pim sparse-mode
```

clear ipv6 mroute pim sparse-mode

Use this command to clear all multicast route table entries learned through PIM-SMv6 for a specified multicast group address, and optionally a specified multicast source address.

Note Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Syntax

```
clear ipv6 mroute <Group-IPv6-add> pim sparse-mode

clear ipv6 mroute <Group-IPv6-add> <Source-IPv6-add> pim sparse-mode
```

Parameter	Description
<Group-IPv6-add>	Specify a multicast group IPv6 address, entered in the form X:X::X:X.
<Source-IPv6-add>	Specify a source group IPv6 address, entered in the form X:X::X:X.

Mode Privileged Exec

Example

```
awplus# clear ipv6 mroute 2001:db8:: pim sparse-mode

awplus# clear ipv6 mroute 2001:db8:: 2002:db8:: pim
sparse-mode
```

clear ipv6 pim sparse-mode bsr rp-set *

Use this command to clear all Rendezvous Point (RP) sets learned through the PIM-SMv6 Bootstrap Router (BSR).

Note Static IPv6 multicast routes take priority over dynamic IPv6 multicast routes. Use the **clear ipv6 mroute** command to clear static IPv6 multicast routes and ensure dynamic IPv6 multicast routes can take over from previous static IPv6 multicast routes.

Syntax

```
clear ipv6 pim sparse-mode bsr rp-set *
```

Parameter	Description
*	Clears all RP sets.

Mode Privileged Exec

Usage For multicast clients, note that one router will be automatically or statically designated as the RP, and all routers must explicitly join through the RP. A Designated Router (DR) sends periodic Join/Prune messages toward a group-specific RP for each group that it has active members.

For multicast sources, note that the Designated Router (DR) unicasts Register messages to the RP encapsulating the data packets from the multicast source. The RP forwards decapsulated data packets toward group members.

Example

```
awplus# clear ipv6 pim sparse-mode bsr rp-set *
```

debug ipv6 pim sparse-mode

Use this command to activate PIM-SMv6 debugging.

Use the **no** variant of this command to deactivate PIMv6 debugging. Note that the **undebug ipv6 pim sparse-mode** command is an alias of the **no** variant of this command.

Syntax

```
debug ipv6 pim sparse-mode [all] [events] [mfc] [mib] [nexthop] [nsm]
    [state] [timer]

no debug ipv6 pim sparse-mode [all] [events] [mfc] [mib] [nexthop]
    [nsm] [state] [timer]
```

Parameter	Description
`all`	Activates/deactivates all PIM-SMv6 debugging.
`events`	Activates debug printing of PIM-SMv6 events.
`mfc`	Activates debug printing of MFC (Multicast Forwarding Cache).
`mib`	Activates debug printing of PIM-SMv6 MIBs.
`nexthop`	Activates debug printing of PIM-SMv6 next hop communications.
`nsm`	Activates debugging of PIM-SMv6 NSM (Network Services Module) communications.
`state`	Activates debug printing of state transition on all PIM-SMv6 FSMs.
`timer`	Activates debug printing of PIM-SMv6 timers.

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# debug ipv6 pim sparse-mode all

awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# debug ipv6 pim sparse-mode events

awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# debug ipv6 pim sparse-mode nexthop
```

Validation output

Figure 53-1: Example output from the show debugging ipv6 pim sparse-mode command after issuing multiple debug ipv6 pim sparse-mode commands

```
awplus#debug ipv6 pim sparse-mode state
awplus#debug ipv6 pim sparse-mode events
awplus#debug ipv6 pim sparse-mode packet
awplus#show debugging ipv6 pim sparse-mode
PIM-SMv6 debugging status:
  PIM event debugging is on
  PIM MFC debugging is off
  PIM state debugging is on
  PIM packet debugging is on
  PIM Hello HT timer debugging is off
  PIM Hello NLT timer debugging is off
  PIM Hello THT timer debugging is off
  PIM Join/Prune JT timer debugging is off
  PIM Join/Prune ET timer debugging is off
  PIM Join/Prune PPT timer debugging is off
  PIM Join/Prune KAT timer debugging is off
  PIM Join/Prune OT timer debugging is off
  PIM Assert AT timer debugging is off
  PIM Register RST timer debugging is off
  PIM Bootstrap BST timer debugging is off
  PIM Bootstrap CRP timer debugging is off
  PIM mib debugging is off
  PIM nsm debugging is off
  PIM nexthop debugging is off
```

Related commands

show debugging ipv6 pim sparse-mode
undebug all ipv6 pim sparse-mode
undebug ipv6 pim sparse-mode

debug ipv6 pim sparse-mode packet

Use this command to activate PIM-SMv6 packet debugging.

Use the **no** variant of this command to deactivate PIMv6 packet debugging.

Syntax

```
debug ipv6 pim sparse-mode packet {in|out}
no debug ipv6 pim sparse-mode packet {in|out}
```

Parameter	Description
`packet`	Activates debug printing of incoming and/or outgoing IPv6 packets.
`in`	Specify incoming packet debugging.
`out`	Specify outgoing packet debugging.

Mode Privileged Exec and Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# debug ipv6 pim sparse-mode packet in

        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# debug ipv6 pim sparse-mode packet out

        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# no debug ipv6 pim sparse-mode packet in

        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# no debug ipv6 pim sparse-mode packet out
```

Related commands **show debugging ipv6 pim sparse-mode**
undebug all ipv6 pim sparse-mode

debug ipv6 pim sparse-mode timer

Use this command to enable debugging for the specified PIM-SMv6 timers.

Use the **no** variants of this command to disable debugging for the specified PIM-SMv6 timers.

Syntax

```
debug ipv6 pim sparse-mode timer assert [at]
no debug ipv6 pim sparse-mode timer assert [at]
debug pim ipv6 sparse-mode timer bsr [bst|crp]
no debug pim ipv6 sparse-mode timer bsr [bst|crp]
debug pim ipv6 sparse-mode timer hello [ht|nlt|tht]
no debug pim ipv6 sparse-mode timer hello [ht|nlt|tht]
debug pim ipv6 sparse-mode timer joinprune [jt|et|ppt|kat|ot]
no debug pim ipv6 sparse-mode timer joinprune [jt|et|ppt|kat|ot]
debug pim ipv6 sparse-mode timer register [rst]
no debug pim ipv6 sparse-mode timer register [rst]
```

Parameter	Description
`assert`	Enable or disable debugging for the Assert timers.
`at`	Enable or disable debugging for the Assert Timer.
`bsr`	Enable or disable debugging for the specified Bootstrap Router timer, or all Bootstrap Router timers.
`bst`	Enable or disable debugging for the Bootstrap Router: Bootstrap Timer.
`crp`	Enable or disable debugging for the Bootstrap Router: Candidate-RP Timer.
`hello`	Enable or disable debugging for the specified Hello timer, or all Hello timers.
`ht`	Enable or disable debugging for the Hello timer: Hello Timer.
`nlt`	Enable or disable debugging for the Hello timer: Neighbor Liveness Timer.
`tht`	Enable or disable debugging for the Hello timer: Triggered Hello Timer.
`joinprune`	Enable or disable debugging for the specified JoinPrune timer, or all JoinPrune timers.
`jt`	Enable or disable debugging for the JoinPrune timer: upstream Join Timer.
`et`	Enable or disable debugging for the JoinPrune timer: Expiry Timer.
`ppt`	Enable or disable debugging for the JoinPrune timer: PrunePending Timer.
`kat`	Enable or disable debugging for the JoinPrune timer: KeepAlive Timer.

Parameter	Description
`ot`	Enable or disable debugging for the JoinPrune timer: Upstream Override Timer.
`register`	Enable or disable debugging for the Register timers.
`rst`	Enable or disable debugging for the Register timer: Register Stop Timer.

Default By default, all debugging is disabled.

Mode Privileged Exec and Global Configuration

Examples To enable debugging for the PIM-SMv6 Bootstrap Router bootstrap timer, use the commands:

```
awplus(config)# debug ipv6 pim sparse-mode timer bsr bst
```

To enable debugging for the PIM-SMv6 Hello: neighbor liveness timer, use the command:

```
awplus(config)# debug ipv6 pim sparse-mode timer hello ht
```

To enable debugging for the PIM-SMv6 Joinprune expiry timer, use the command:

```
awplus# debug ipv6 pim sparse-mode timer joinprune et
```

To disable debugging for the PIM-SMv6 Register timer, use the command:

```
awplus# no debug ipv6 pim sparse-mode timer register
```

Related commands **show debugging ipv6 pim sparse-mode**

ipv6 pim accept-register

Use this command to configure the ability to filter out multicast sources specified by the given software IPv6 access-list at the Rendezvous Point (RP), so that the RP will accept/ refuse to perform the register mechanism for the packets sent by the specified sources. By default, the RP accepts register packets from all multicast sources.

Use the **no** variant of this command to revert to default.

Syntax

```
ipv6 pim accept-register list{<access-list>}
no ipv6 pim accept-register
```

Parameter	Description
<access-list>	Specify a Standard or an Extended software IPv6 Access list. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim accept-register list G2
awplus(config)# ipv6 access-list standard G2 permit 2001:db8::/
128

awplus# configure terminal
awplus(config)# no ipv6 pim accept-register
```

ipv6 pim anycast-rp

Use this command to configure Anycast RP (Rendezvous Point) in an RP set.

Use the **no** variant of this command to remove the configuration.

Syntax

```
ipv6 pim anycast-rp <anycast-rp-address> <member-rp-address>
no ipv6 pim anycast-rp <anycast-rp-address> [<member-rp-address>]
```

Parameter	Description
<anycast-rp-address>	<X:X::X:X> Specify an Anycast IPv6 address to configure an Anycast RP (Rendezvous Point) in a RP set.
<member-rp-address>	<A:B::C:D> Specify an Anycast RP (Rendezvous Point)IPv6 address to configure an Anycast RP in a RP set.

Mode Global Configuration

Usage Anycast is a network addressing and routing scheme where data is routed to the nearest or best destination as viewed by the routing topology. Compared to unicast with a one-to-one association between network address and network endpoint, and multicast with a one-to-many association between network address and network endpoint; anycast has a one-to-many association between network address and network endpoint. For anycast, each destination address identifies a set of receiver endpoints, from which only one receiver endpoint is chosen.

Use this command to specify the Anycast RP configuration in the Anycast RP set. Use the **no** variant of this command to remove the Anycast RP configuration. Note that the member RP address is optional when using the **no** parameter to remove the Anycast RP configuration. removing the anycast RP address also removes the member RP address.

Examples The following example shows how to configure the Anycast RP address with **ipv6 pim anycast-rp**:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim anycast-rp 2:2::2:2 20:20::20:20
```

The following example shows how to remove the Anycast RP in the RP set specifying only the anycast RP address with **no ipv6 pim anycast-rp**, but not specifying the member RP address:

```
awplus# configure terminal
awplus(config)# no ipv6 pim anycast-rp 2:2::2:2 20:20::20:20
```

ipv6 pim bsr-border

Use the **ipv6 pim bsr-border** command to prevent Bootstrap Router (BSR) messages from being sent or received through a VLAN interface. The BSR border is the border of the PIM-SMv6 domain.

Use the **no** variant of this command to disable the configuration set with **ipv6 pim bsr-border**.

Syntax

```
ipv6 pim bsr-border
no ipv6 pim bsr-border
```

Mode Interface Configuration for a VLAN interface.

Usage When this command is configured on a VLAN interface, no PIM-SMv6 BSR messages will be sent or received through the interface. Configure an interface bordering another PIM-SMv6 domain with this command to avoid BSR messages from being exchanged between the two PIM-SMv6 domains.

BSR messages should not be exchanged between different domains, because devices in one domain may elect Rendezvous Points (RPs) in the other domain, resulting in loss of isolation between the two PIM domains that would stop the PIM-SMv6 protocol from working as intended.

Examples The following example configures the VLAN interface `vlan2` to be the PIM-SMv6 domain border:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim bsr-border
```

The following example removes the VLAN interface `vlan2` from the PIM-SMv6 domain border:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim bsr-border
```

ipv6 pim bsr-candidate

Use this command to give the device the candidate BSR (Bootstrap Router) status using the specified IPv6 address mask of the interface.

Use the **no** variant of this command to withdraw the address of the interface from being offered as a BSR candidate.

Syntax

```
ipv6 pim bsr-candidate <interface> [<hash>] [<priority>]
no ipv6 pim bsr-candidate [<interface>]
```

Parameter	Description
<interface>	Specify the interface. For instance, VLAN interface `vlan2`.
<hash>	`<0-128>` configure the hash mask length used for RP selection. The default hash value if you do not configure this parameter is 126.
<priority>	`<0-255>` configure priority for a BSR candidate. Note that you must also specify the <*hash*> (mask length) when specifying the <*priority*>. The default priority if you do not configure this parameter is 64.

Mode Global Configuration

Default The default hash parameter value is 126 and the default priority parameter value is 64.

Examples To set the BSR candidate to the VLAN interface `vlan2`, with the optional mask length and BSR priority parameters, enter the commands shown below:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim bsr-candidate vlan2 20 30
```

To withdraw the address of `vlan2` from being offered as a BSR candidate, enter:

```
awplus# configure terminal
awplus(config)# no ipv6 pim bsr-candidate vlan2
```

ipv6 pim cisco-register-checksum

Use this command to configure the option to calculate the Register Checksum over the whole packet. This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to disable this option.

Syntax

```
ipv6 pim cisco-register-checksum
no ipv6 pim cisco-register-checksum
```

Default This command is disabled by default. By default, Register Checksum is calculated only over the header.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim cisco-register-checksum

awplus# configure terminal
awplus(config)# no ipv6 pim cisco-register-checksum
```

ipv6 pim cisco-register-checksum group-list

Use this command to configure the option to calculate the Register Checksum over the whole packet on multicast groups as specified by the software IPv6 access-list. This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to revert to default settings.

Syntax

```
ipv6 pim cisco-register-checksum group-list <IPv6-access-list>
no ipv6 pim cisco-register-checksum group-list <IPv6-access-list>
```

Parameter	Description
<IPv6-access-list>	Optional. Specify a Standard or Extended software IPv6 access list. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs. Use this parameter to configure the option to calculate the Register Checksum over the whole packet on multicast groups as specified by an IPv6 access list entered after this command.

Mode Global Configuration

Default This command is disabled by default. By default, Register Checksum is calculated only over the header.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim cisco-register-checksum group-list G1
awplus(config)# ipv6 access-list standard G1 permit
ff0x::db8:0:0/96
```

ipv6 pim crp-cisco-prefix

Use this command to interoperate with Cisco devices that conform to an earlier draft standard. Some Cisco devices might not accept candidate RPs with a group prefix number of zero. Note that the latest BSR specification prohibits sending RP advertisements with prefix 0.

Use the **no** variant of this command to revert to the default settings.

Syntax

```
ipv6 pim crp-cisco-prefix
no ipv6 pim crp-cisco-prefix
```

Mode Global Configuration

Usage Cisco's BSR code does not conform to the latest BSR draft, it does not accept candidate RPs with a group prefix number of zero. To make the candidate RP work with a Cisco BSR, use the **ipv6 pim crp-cisco-prefix** command when interoperating with older versions of Cisco IOS.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim crp-cisco-prefix

awplus# configure terminal
awplus(config)# no ipv6 pim crp-cisco-prefix
```

Related commands ipv6 pim rp-candidate

ipv6 pim dr-priority

Use this command to set the Designated Router priority value.

Use the **no** variant of this command to disable this function.

Syntax

```
ipv6 pim dr-priority  <priority>
no ipv6 pim dr-priority [<priority>]
```

Parameter	Description
<priority>	<0-4294967294> Specify the Designated Router priority value. Note that a higher value has a higher preference or higher priority.

Default The default value is 1. The negated form of this command restores the value to the default.

Mode Interface Configuration for a VLAN interface.

Examples To set the Designated Router priority value to `11234` for the VLAN interface `vlan2`, apply the commands as shown below:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim dr-priority 11234
```

To disable the Designated Router priority value for the VLAN interface `vlan2`, apply the commands as shown below:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim dr-priority
```

Related commands **ipv6 pim ignore-rp-set-priority**

ipv6 pim exclude-genid

Use this command to exclude the GenID option from Hello packets sent out by the PIM-SMv6 module on a particular interface. This command is used to inter-operate with older Cisco IOS versions.

Use the **no** variant of this command to revert to default settings.

Syntax

```
ipv6 pim exclude-genid
no ipv6 pim exclude-genid
```

Default By default, this command is disabled; the GenID option is included.

Mode Interface Configuration for a VLAN interface.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim exclude-genid
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim exclude-genid
```

ipv6 pim ext-srcs-directly-connected

Use this command to configure PIM-SMv6 to treat all source traffic arriving on the interface as though it was sent from a host directly connected to the interface.

Use the **no** variant of this command to configure PIM-SMv6 to treat only directly connected sources as directly connected.

Syntax

```
ipv6 pim ext-srcs-directly-connected
no ipv6 pim ext-srcs-directly-connected
```

Default The **no** variant of this command is the default behavior.

Mode Interface Configuration for a VLAN interface.

Example To configure PIM-SMv6 to treat all sources as directly connected for VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim ext-srcs-directly-connected
```

To configure PIM-SMv6 to treat only directly connected sources as directly connected for VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim ext-srcs-directly-connected
```

ipv6 pim hello-holdtime

This command configures a hello-holdtime value. You cannot configure a hello-holdtime value that is less than the current hello-interval.

Use the **no** variant of this command to return it to its default of 3.5 * the current hello-interval.

Syntax

```
ipv6 pim hello-holdtime <holdtime>
no ipv6 pim hello-holdtime
```

Parameter	Description
<holdtime>	<1-65535> The holdtime value in seconds (no fractional seconds are accepted).

Default The default hello-holdtime value is 3.5 * the current hello-interval. The default hello-holdtime is restored using the negated form of this command.

Mode Interface Configuration for a VLAN interface.

Usage Each time the hello interval is updated, the hello holdtime is also updated, according to the following rules:

If the hello holdtime is not configured; or if the hello holdtime is configured and less than the current hello-interval value, it is modified to the (3.5 * hello interval). Otherwise, it retains the configured value.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim hello-holdtime 123
```

ipv6 pim hello-interval

This command configures a hello-interval value for PIM-SMv6.

Use the **no** variant of this command to reset the hello-interval for PIM-SMv6 to the default.

Syntax

```
ipv6 pim hello-interval <interval>
no ipv6 pim hello-interval
```

Parameter	Description
<interval>	<1-65535> The value in seconds (no fractional seconds accepted).

Default The default hello-interval value is 30 seconds. The default is restored using the negated form of this command.

Mode Interface Configuration for a VLAN interface.

Usage When the hello interval is configured, and the hello holdtime is not configured, or when the configured hello-holdtime value is less than the new hello-interval value; the holdtime value is modified to the (3.5 * hello interval). Otherwise, the hello-holdtime value is the configured value.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim hello-interval 123
```

ipv6 pim ignore-rp-set-priority

Use this command to ignore the RP-SET priority value, and use only the hashing mechanism for RP selection.

Use the **no** variant of this command to disable this setting.

Syntax

```
ipv6 pim ignore-rp-set-priority
no ipv6 pim ignore-rp-set-priority
```

Mode Global Configuration

Usage This command is used to inter-operate with older Cisco IOS versions.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim ignore-rp-set-priority

awplus# configure terminal
awplus(config)# no ipv6 pim ignore-rp-set-priority
```

ipv6 pim jp-timer

Use this command to set the PIM-SMv6 join/prune timer. Note that the value set by the join/prune timer is the value that the switch puts into the holdtime field of the join/prune packets it sends to its neighbors.

Use the **no** variant of this command to return the PIM-SMv6 join/prune timer to its default value of 210 seconds.

Syntax

```
ipv6 pim jp-timer <1-65535>
no ipv6 pim jp-timer [<1-65535>]
```

Parameter	Description
<1-65535>	Specifies the Join/Prune timer value. The default value is 210 seconds.

Default The default PIM-SMv6 join/prune timer value is 210 seconds.

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim jp-timer 300

awplus# configure terminal
awplus(config)# no ipv6 pim jp-timer
```

ipv6 pim neighbor-filter

This command enables filtering of neighbors on the VLAN interface. When configuring a neighbor filter, PIM-SMv6 will either not establish adjacency with the neighbor, or terminate adjacency with the existing neighbors if denied by the filtering IPv6 access list.

Use the **no** variant of this command to disable this function.

Syntax

```
ipv6 pim neighbor-filter <IPv6-accesslist>
no ipv6 pim neighbor-filter <IPv6-accesslist>
```

Parameter	Description
<IPv6-accesslist>	Specify a Standard or an Extended software IPv6 access list name for the PIM-SMv6 neighbor filter. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Default By default, there is no neighbor filtering applied to an interface.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config)# ipv6 enable
awplus(config-if)# ipv6 pim neighbor-filter filter1
awplus(config-if)# ipv6 access-list standard filter1 deny
fe80:20e:cff:fe01:facc
awplus(config-if)# ipv6 access-list standard filter1 permit
any
awplus(config-if)# exit
```

ipv6 pim register-rate-limit

Use this command to configure the rate of register packets sent by this DR, in units of packets per second. The configured rate is per (S, G) state, and is not a system wide rate.

Use the **no** variant of this command to remove the limit and reset to the default rate limit.

Syntax

```
ipv6 pim register-rate-limit <1-65535>
no ipv6 pim register-rate-limit
```

Parameter	Description
<1-65535>	Specifies the maximum number of packets that can be sent per second.

Mode Global Configuration

Default The default is 0, as reset with the **no** variant, which also specifies an unlimited rate limit.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim register-rate-limit 3444

awplus# configure terminal
awplus(config)# no ipv6 pim register-rate-limit 3444
```

ipv6 pim register-rp-reachability

Use this command to enable the RP reachability check for PIMv6 Register processing at the DR. The default setting is no checking for RP-reachability.

Use the **no** variant of this command to disable this processing.

Syntax

```
ipv6 pim register-rp-reachability
no ipv6 pim register-rp-reachability
```

Default This command is disabled; by default, there is no checking for RP-reachability.

Mode Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim register-rp-reachability

awplus# configure terminal
awplus(config)# no ipv6 pim register-rp-reachability
```

ipv6 pim register-source

Use this command to configure the source IPv6 address of register packets sent by this DR, overriding the default source IPv6 address, which is the IPv6 address of the RPF interface toward the source host.

Use the **no** variant of this command to remove the IPv6 source address of Register packets sent by this DR, reverting back to use the default IPv6 source address that is the address of the RPF interface toward the source host.

Syntax

```
ipv6 pim register-source [<source-IPv6-address>|<interface>]
no ipv6 pim register-source
```

Parameter	Description
<source-IPv6-address>	The IPv6 address, entered in the form `X:X::X:X`, to be used as the source of the register packets.
<interface>	The name of the VLAN interface to be used as the source of the register packets.

Usage The configured address must be a reachable address to be used by the RP to send corresponding Register-Stop messages in response. It is normally the local loopback IPv6 interface address, but can also be a physical IPv6 address. This IPv6 address must be advertised by unicast routing protocols on the DR. The configured interface does not have to be PIM-SMv6 enabled.

Mode Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim register-source 3ffe::24:2

awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim register-source vlan2

awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# no ipv6 pim register-source
```

ipv6 pim register-suppression

Use this command to configure the register-suppression time, in seconds, overriding the default of 60 seconds.

Use the **no** variant of this command to reset the value to its default of 60 seconds.

Syntax

```
ipv6 pim register-suppression <1-65535>
no ipv6 pim register-suppression
```

Parameter	Description
<1-65535>	Register suppression on time in seconds.

Mode Global Configuration

Default The default PIM-SMv6 register suppression time is 60 seconds, and is restored with the no variant of this command.

Usage Configuring this value modifies register-suppression time at the DR. Configuring this value at the RP modifies the RP-keepalive-period value if the **ipv6 pim rp-register-kat** command on page 53.33 is not used.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim register-suppression 192

awplus# configure terminal
awplus(config)# no ipv6 pim register-suppression
```

ipv6 pim rp-address

Use this command to statically configure RP (Rendezvous Point) address for IPv6 multicast groups.

Use the **no** variant of this command to remove a statically configured RP (Rendezvous Point) address for IPv6 multicast groups.

Syntax

```
ipv6 pimv6 rp-address <IPv6-address> [<IPv6-access-list>] [override]
no ipv6 pim rp-address <IPv6-address> [<IPv6-access-list>] [override]
```

Parameter	Description
`<IPv6-address>`	Specify the IPv6 address of the Rendezvous Point, entered in the form `X:X::X:X`.
`<IPv6-access-list>`	Specify a Standard or an Extended software IPv6 access-list name. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.
`override`	Specify this optional parameter keyword to enable any statically defined RPs to override dynamically learned RPs.

Mode Global Configuration

Usage The AlliedWare Plus™ PIM-SMv6 implementation supports multiple static RPs. It also supports usage of static-RP and BSR mechanism simultaneously. The **ipv6 pim rp-address** command is used to statically configure the RP address for IPv6 multicast groups.

You need to understand the following information before using this command.

If the RP-address that is configured by the BSR, and the RP-address that is configured statically, are both available for a group range, then the RP-address configured through BSR is chosen over the statically configured RP-address.

A single static-RP can be configured for multiple group ranges using software IPv6 access-lists (ACLs). However, configuring multiple static RPs (using **ipv6 pim rp-address** command) with the same RP address is not allowed. The static-RP can either be configured for the whole multicast group range `ff00::/8` (without using IPv6 ACLs) or for specific group ranges (when using IPv6 ACLs).

For example, configuring **ipv6 pim rp-address 3ffe:10:10:5::153** will configure static-RP `3ffe:10:10:5::153` for the default group range `ff00::/8`. Configuring **ipv6 pim rp-address 3ffee:20:20:5::153 grp-list** will configure static-RP `3ffe:10:10:5::153` for all the group ranges represented by permit filters in the defined named **grp-list** ACL.

If multiple static-RPs are available for a group range, then one with the highest IPv6 address is chosen.

Only `permit` filters in IPv6 ACL are considered as valid group ranges. The default `permit` filter `::/0` is converted to the default multicast filter `ff00::/8`.

After configuration, the RP-address is inserted into a static-RP group tree based on the configured group ranges. For each group range, multiple static-RPs are maintained in a list. This list is sorted in a descending order of IPv6 addresses. When selecting static-RPs for a group range, the first element (which is the static-RP with highest IPv6 address) is chosen.

RP-address deletion is handled by removing the static-RP from all the existing group ranges and recalculating the RPs for existing TIB states if required.

Group mode and RP address mappings learned through BSR take precedence over mappings statistically defined by the **ipv6 pim rp-address** command. Commands with the **override** keyword take precedence over dynamically learned mappings.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 access-list standard G2 permit
2001:db8::/128
awplus(config)# ipv6 pim rp-address 3ffe:30:30:5::153 G2

awplus# configure terminal
awplus(config)# no ipv6 pim rp-address 3ffe:30:30:5::153 G2
```

Related commands

ipv6 pim rp-candidate
ipv6 pim rp-register-kat

ipv6 pim rp-candidate

Use this command to give the switch the candidate RP (Rendezvous Point) status using the IPv6 address of the specified VLAN interface.

Use the **no** variant of this command to remove the RP status set using the **ipv6 pim rp-candidate** command.

Syntax

```
ipv6 pim rp-candidate <interface> [priority <priority>|
   interval <interval>| grouplist <accesslist>]

no ipv6 pim rp-candidate [<interface>]
```

Parameter	Description
<interface>	Specify a VLAN interface name.
<priority>	<0-255> Specify this to configure the priority for an RP candidate.
<interval>	Specify a candidate RP advertisement interval in the range <1-16383> (seconds).
<accesslist>	Specify a Standard or an Extended software IPv6 access list name. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Default The priority value for a candidate RP is 192 by default until specified using the **priority** parameter.

Mode Global Configuration

Usage Note that issuing the command **ipv6 pim rp-candidate** *<interface>* without optional **priority**, **interval**, or **grouplist** parameters will configure the candidate RP with a priority value of 192.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim rp-candidate vlan2 priority 3
```

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 access-list standard G2 permit 2001:db8::/
128
awplus(config)# ipv6 pim rp-candidate vlan2 priority 3
group-list G2
```

```
awplus# configure terminal
awplus(config)# no ipv6 pim rp-candidate vlan2
```

Related commands **ipv6 pim rp-address**
ipv6 pim rp-register-kat

ipv6 pim rp embedded

Use this command to configure and enable embedded RP (Rendezvous Point) in PIM-SMv6.

Note: This command only applies to the embedded RP group range **ff7x::/12** and **fffx::/12**.

Use the **no** variant of this command to disable embedded RP support. Since embedded RP support is enabled by default, use the **no** variant of this command to disable the default.

Syntax

```
ipv6 pim rp embedded
no ipv6 pim rp embedded
```

Mode Global Configuration

Default Embedded RP is enabled by default in the AlliedWare Plus implementation of PIM-SMv6.

Examples The following example re-enables embedded RP support, the default state in PIM-SMv6:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim rp embedded
```

The following example disables embedded RP support, which is enabled by default in PIM-SMv6:

```
awplus# configure terminal
awplus(config)# no ipv6 pim rp embedded
```

ipv6 pim rp-register-kat

Use this command to configure the Keep Alive Time (KAT) for (S,G) states at the RP (Rendezvous Point) to monitor PIM-SMv6 Register packets.

Use the **no** variant of this command to return the PIM-SMv6 KAT timer to its default value of 210 seconds.

Syntax

```
ipv6 pim rp-register-kat <1-65535>
no ipv6 pim rp-register-kat
```

Parameter	Description
<1-65536>	Specify the KAT timer in seconds. The default value is 210 seconds.

Mode Global Configuration

Default The default PIM-SMv6 KAT timer value is 210 seconds.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim rp-register-kat 3454

awplus# configure terminal
awplus(config)# no ipv6 pim rp-register-kat
```

Related commands **ipv6 pim rp-address**
ipv6 pim rp-candidate

ipv6 pim sparse-mode

Use this command to enable PIM-SMv6 on a VLAN interface.

Use the **no** variant of this command to disable PIM-SMv6 on a VLAN interface.

Syntax

```
ipv6 pim sparse-mode
no ipv6 pim sparse-mode
```

Mode Interface Configuration for a VLAN interface.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim sparse-mode
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim sparse-mode
```

ipv6 pim sparse-mode passive

Use this command to enable and disable PIM-SMv6 passive mode operation for local members on a VLAN interface.

Use the **no** variant of this command to disable PIM-SMv6 passive mode operation for local members on a VLAN interface.

Syntax

```
ipv6 pim sparse-mode passive
no ipv6 pim sparse-mode passive
```

Mode Interface Configuration for a VLAN interface.

Usage Passive mode essentially stops PIM-SMv6 transactions on the interface, allowing only the MLD mechanism to be active.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim sparse-mode passive
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim sparse-mode passive
```

ipv6 pim spt-threshold

This command turns on the ability for the last-hop PIM-SMv6 router to switch to SPT.

The **no** variant of this command turns off the ability for the last-hop PIM-SMv6 router to switch to SPT.

Note The switching to SPT happens either at the receiving of the first data packet, or not at all; it is not rate-based.

Syntax

```
ipv6 pim spt-threshold
no ipv6 pim spt-threshold
```

Mode Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim spt-threshold

awplus# configure terminal
awplus(config)# no ipv6 pim spt-threshold
```

ipv6 pim spt-threshold group-list

Use this command to turn on/off the ability for the last-hop PIM-SMv6 router to switch to SPT for multicast group addresses as specified by the given software IPv6 access-list.

Use the **no** variant of this command to turn off switching to the SPT.

Note The switching to SPT happens either at the receiving of the first data packet, or not at all; it is not rate-based.

Syntax

```
ipv6 pim spt-threshold group-list <IPv6-access-list>
no ipv6 pim spt-threshold group-list <IPv6-access-list>
```

Parameter	Description
<IPv6-access-list>	Specify a Standard or an Extended software IPv6 access-list name. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Mode Global Configuration

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 pim spt-threshold group-list G1
awplus(config)# ipv6 access-list standard G1 permit
2001:db8::/128

awplus# configure terminal
awplus(config)# no ipv6 pim spt-threshold group-list G1
```

ipv6 pim unicast-bsm

Use this command to enable support for the sending and receiving of unicast Boot Strap Messages (BSM) on a VLAN interface.

Use the **no** variant of this command to disable the sending and receiving of unicast BSM on a VLAN interface.

Syntax

```
ipv6 pim unicast-bsm
no ipv6 pim unicast-bsm
```

Mode Interface Configuration for a VLAN interface.

Default Unicast BSM is disabled by default on an interface.

Usage This command provides backward compatibility with older versions of the Boot Strap Router (BSR) specification, which directs unicast BSM to refresh the state of new or restarting neighbors. The current BSR specification defines a No Forward BSM to achieve the same result.

Examples

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 pim unicast-bsm

awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 pim unicast-bsm
```

show debugging ipv6 pim sparse-mode

This command displays the status of the PIM-SMv6 debugging on your switch.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show debugging ipv6 pim sparse-mode`

Mode User Exec and Privileged Exec

Example To display PIM-SMv6 debugging settings, use the command:

```
awplus# show debugging ipv6 pim sparse-mode
```

Figure 53-2: Example output from the show debugging ipv6 pim sparse-mode command

```
awplus#show debugging ipv6 pim sparse-mode
Debugging status:
  PIM event debugging is on
  PIM MFC debugging is on
  PIM state debugging is on
  PIM packet debugging is on
  PIM Hello HT timer debugging is on
  PIM Hello NLT timer debugging is on
  PIM Hello THT timer debugging is on
  PIM Join/Prune JT timer debugging is on
  PIM Join/Prune ET timer debugging is on
  PIM Join/Prune PPT timer debugging is on
  PIM Join/Prune KAT timer debugging is on
  PIM Join/Prune OT timer debugging is on
  PIM Assert AT timer debugging is on
  PIM Register RST timer debugging is on
  PIM Bootstrap BST timer debugging is on
  PIM Bootstrap CRP timer debugging is on
```

Related commands **debug ipv6 pim sparse-mode**
undebug ipv6 pim sparse-mode

show ipv6 pim sparse-mode bsr-router

Use this command to show the PIM-SMv6 Bootstrap Router (BSR) IPv6 address.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode bsr-router`

Mode User Exec and Privileged Exec

Example To display the BSR IPv6 address, use the command:

```
awplus# show ipv6 pim sparse-mode bsr-router
```

Output **Figure 53-3: Example output from the show ipv6 pim sparse-mode bsr-router command**

```
awplus#show ipv6 pim sparse-mode bsr-router
PIM6v2 Bootstrap information
  BSR address: 2001:203::213 (?)
  Uptime:      00:36:25, BSR Priority: 64, Hash mask length: 126
  Expires:     00:01:46
  Role: Candidate BSR
  State: Candidate BSR

  Candidate RP: 2001:5::211(vlan5)
    Advertisement interval 60 seconds
    Next C-RP advertisement in 00:00:43
```

Related commands **show ipv6 pim sparse-mode rp mapping**
show ipv6 pim sparse-mode neighbor

show ipv6 pim sparse-mode interface

Use this command to show PIM-SMv6 interface information. Note that you can specify an individual VLAN interface with the optional parameter. Alternatively, you can display PIM-SMv6 interface information for all interfaces if you omit the optional interface parameter.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode interface`

Mode User Exec and Privileged Exec

Examples To display information about all PIM-SMv6 interfaces, use the command:

```
awplus# show ipv6 pim sparse-mode interface
```

```
awplus#show ipv6 pim sparse-mode interface
Interface VIFindex Ver/   Nbr    DR
                   Mode   Count  Priority
vlan2     0        v2/S   2      1
    Address       : fe80::207:e9ff:fe02:81d
    Global Address: 3ffe:192:168:1::53
    DR            : fe80::20e:cff:fe01:facc
vlan3     2        v2/S   2      1
    Address       : fe80::207:e9ff:fe02:21a2
    Global Address: 3ffe:192:168:10::53
    DR            : this system
```

Table 53-1: Parameters in the output from the show ipv6 pim sparse-mode interface command

Parameters	Description
`Address`	Primary PIM-SMv6 address.
`Interface`	Name of the PIM-SMv6 interface.
`VIF Index`	The Virtual Interface index of the VLAN.
`Ver/Mode`	PIMv6 version/Sparse mode.
`Nbr Count`	Neighbor count of the PIM-SMv6 interface.
`DR Priority`	Designated Router priority.
`DR`	The IPv6 address of the Designated Router.

Related commands
ipv6 pim sparse-mode
show ipv6 pim sparse-mode rp mapping
show ipv6 pim sparse-mode neighbor

show ipv6 pim sparse-mode interface detail

Use this command to show detailed PIM-SMv6 information for all PIM-SMv6 configured interfaces.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode interface detail`

Mode User Exec and Privileged Exec

Example To show detailed PIM-SMv6 information for all PIM-SMv6 configured interfaces, use the command:

```
awplus# show ipv6 pim sparse-mode interface detail
```

Output **Figure 53-4: Example output from the show ipv6 pim sparse-mode interface detail command**

```
awplus#show ipv6 pim sparse-mode interface detail
vlan2 (vif 0)
  Address fe80::207:e9ff:fe02:81d, DR fe80::20e:cff:fe01:facc
  Hello period 30 seconds, Next Hello in 21 seconds
  Triggered Hello period 5 seconds
  Secondary addresses:
   3ffe:192:168:1::53
  Neighbors:
   fe80::202:b3ff:fed4:69fe
   fe80::20e:cff:fe01:facc

vlan3 (vif 2):
  Address fe80::207:e9ff:fe02:21a2, DR fe80::207:e9ff:fe02:21a2
  Hello period 30 seconds, Next Hello in 20 seconds
  Triggered Hello period 5 seconds
  Secondary addresses:
   3ffe:192:168:10::53
  Neighbors:
```

show ipv6 pim sparse-mode local-members

Use this command to show detailed local member information on a VLAN interface configured for PIM-SMv6. If you do not specify a VLAN interface then detailed local member information is shown for all VLAN interfaces configured for PIM-SMv6.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode local-members [<interface>]`

Parameter	Description
<interface>	Optional Specify the interface. For instance, VLAN interface `vlan2`.

Mode User Exec and Privileged Exec

Example To show detailed PIM-SMv6 information for all PIM-SMv6 configured VLAN interfaces, use the command:

```
awplus# show ipv6 pim sparse-mode local-members
```

Output **Figure 53-5: Example output from the show ipv6 pim sparse-mode local-members command**

```
awplus#show ipv6 pim sparse-mode local-members
PIM Local membership information

vlan1:

  (*, ff02::1:ff6b:4783) : Include

vlan203:

  (*, ff0e:1::4) : Include
```

Example To show detailed PIM-SMv6 information for the PIM-SMv6 configured interface `vlan1`, use the command:

```
awplus# show ipv6 pim sparse-mode local-members vlan1
```

Output **Figure 53-6: Example output from the show ipv6 pim sparse-mode local-members vlan1 command**

```
awplus#show ipv6 pim sparse-mode local-members vlan1
PIM Local membership information

vlan1:

  (*, ff02::1:ff6b:4783) : Include
```

show ipv6 pim sparse-mode mroute

This command displays the IPv6 multicast routing table, or the IPv6 multicast routing table based on the specified IPv6 address or addresses.

Two group IPv6 addresses cannot be used simultaneously; two source IPv6 addresses cannot be used simultaneously.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 pim sparse-mode mroute [<group-IPv6-address>|<source-IPv6-address>]

show ipv6 pim sparse-mode mroute [<group-IPv6-address> <source-IPv6-address>]

show ipv6 pim sparse-mode mroute [<source-IPv6-address> <group-IPv6-address>]
```

Parameter	Description
<group-IPv6-address>	Group IPv6 address, entered in the form X:X::X:X. Based on the group and source IPv6 address, the output is the selected route if present in the multicast route tree.
<source-IPv6-address>	Source IPv6 address, entered in the form X:X::X:X. Based on the source and group IPv6 address, the output is the selected route if present in the multicast route tree.

Mode User Exec and Privileged Exec

Usage Note that when a feature license is enabled, the output for the **show ipv6 pim sparse-mode mroute** command will only show 100 interfaces because of the terminal display width limit. Use the **show ipv6 pim sparse-mode mroute detail** command to display detailed entries of the IPv6 multicast routing table.

Examples

```
awplus# show ipv6 pim sparse-mode mroute

awplus# show ipv6 pim sparse-mode mroute 2001:db8::

awplus# show ipv6 pim sparse-mode mroute 2001:db8::
        2002:db8::
```

Figure 53-7: Example output from the show ipv6 pim sparse-mode mroute command

```
awplus#show ipv6 pim sparse-mode mroute
IPv6 Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 2
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 2

(*, ff0x::db8:0:0/96)
RP: 3ffe:10:10:5::153
RPF nbr: fe80::202:b3ff:fed4:69fe
RPF idx: wm0
Upstream State: JOINED
 Local      ..l.........................................
 Joined     ..l.........................................
 Asserted   ..l.........................................
FCR:
Source: 3ffe:10:10:1::96
 Outgoing   ..o.........................................
 KAT timer running, 205 seconds remaining
 Packet count 1

(*, ff0x::db8:0:0/96)
RP: 3ffe:10:10:5::153
RPF nbr: fe80::202:b3ff:fed4:69fe
RPF idx: wm0
Upstream State: JOINED
 Local      ..l.........................................
 Joined     ..l.........................................
 Asserted   ..l.........................................
FCR:
Source: 3ffe:10:10:1::96
 Outgoing   ..o.........................................
 KAT timer running, 208 seconds remaining
 Packet count 1
```

show ipv6 pim sparse-mode mroute detail

This command displays detailed entries of the IPv6 multicast routing table, or detailed entries of the IPv6 multicast routing table based on the specified IPv6 address or addresses.

Two group IPv6 addresses cannot be used simultaneously; two IPv6 source addresses cannot be used simultaneously.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 pim sparse-mode mroute [<group-IPv6-address>|<source-IPv6-address>] detail

show ipv6 pim sparse-mode mroute [<group-IPv6-address> <source-IPv6-address>] detail

show ipv6 pim sparse-mode mroute [<source-IPv6-address> <group-IPv6-address>] detail
```

Parameter	Description
<group-IPv6-address>	Group IPv6 address, entered in the form X:X::X:X. Output is all multicast entries belonging to that group.
<source-IPv6-address>	Source IPv6 address, entered in the form X:X::X:X. Output is all multicast entries belonging to that source.
`detail`	Show detailed information.

Usage

Based on the group and source IPv6 address, the output is the selected route if present in the multicast route tree.

Mode

User Exec and Privileged Exec

Examples

```
awplus# show ipv6 pim sparse-mode mroute detail

awplus# show ipv6 pim sparse-mode mroute 2001:db8:: detail

awplus# show ipv6 pim sparse-mode mroute 2001:db8::
        2002:db8:: detail
```

Figure 53-8: Example output from the show ipv6 pim sparse-mode mroute detail command

```
awplus#show ipv6 pim sparse-mode mroute detail
IPv6 Multicast Routing Table

(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 0

(*, ff13::10) Uptime: 00:00:09
  RP: ::, RPF nbr: None, RPF idx: None
  Upstream:
   State: JOINED, SPT Switch: Enabled, JT: off
   Macro state: Join Desired,
  Downstream:
   vlan2:
     State: NO INFO, ET: off, PPT: off
     Assert State: NO INFO, AT: off
      Winner: ::, Metric: 4294967295l, Pref: 4294967295l, RPT bit: on
     Macro state: Could Assert, Assert Track
  Local Olist:
   vlan3
  FCR:
```

show ipv6 pim sparse-mode neighbor

Use this command to show the PIM-SMv6 neighbor information.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax

```
show ipv6 pim sparse-mode neighbor [<interface>] [<IPv6-address>]
   [detail]
```

Parameter	Description
<interface>	Interface name (e.g. `vlan2`). Show neighbors on an interface.
<IPv6-address>	Show neighbors with a particular address on an interface. The IPv6 address entered in the form X:X::X:X.
`detail`	Show detailed information.

Mode User Exec and Privileged Exec

Examples

```
awplus# show ipv6 pim sparse-mode neighbor
```

```
awplus# show ipv6 pim sparse-mode neighbor vlan5 detail
```

Figure 53-9: Example output from the show ipv6 pim sparse-mode neighbor command

```
awplus#show ipv6 pim sparse-mode neighbor
Neighbor Address                     Interface   Uptime/Expires      DR
                                                                        Pri/Mode
fe80::202:b3ff:fed4:69fe              vlan2       05:33:52/00:01:41 1 /
fe80::20e:cff:fe01:facc               vlan3       05:33:53/00:01:26 1 / DR
```

Figure 53-10: Example output from the show ipv6 pim sparse-mode neighbor interface detail command

```
awplus#show ipv6 pim sparse-mode neighbor detail
Nbr fe80::211:11ff:fe44:4cd8 (vlan1), DR
 Expires in 64 seconds, uptime 00:00:53
 Holdtime: 70 secs, T-bit: off, Lan delay: 1, Override interval: 3
 DR priority: 100, Gen ID: 1080091886,
 Secondary addresses:
  3ffe:10:10:10:3::180
```

show ipv6 pim sparse-mode nexthop

Use this command to see the next hop information as used by PIM-SMv6.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ipv6 pim sparse-mode nexthop`

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 pim sparse-mode nexthop
```

Figure 53-11: Example output from the show ipv6 pim sparse-mode nexthop command

```
awplus#show ipv6 pim sparse-mode nexthop
Flags: N = New, R = RP, S = Source, U = Unreachable
Destination           Type  Nexthop Nexthop Nexthop  Nexthop Metric     Pref    Refcnt
                            Num     Addr    Ifindex  Name
_____________________________________________________________________________________
3ffe:10:10:5::153     .RS.  1       fe80::20e:cff:fe01:facc  2     30   110     1
```

Table 53-2: Parameters in output of the show ipv6 pim sparse-mode nexthop command

Parameter	Description
`Destination`	The destination address for which PIM-SMv6 requires next hop information.
`Type`	The type of destination, as indicated by the Flags description. N = New, R= RP, S = Source, U = Unreachable.
`Nexthop Num`	The number of next hops to the destination. PIM-SMv6 always uses only 1 next hop.
`Nexthop Addr`	The address of the primary next hop gateway.
`Nexthop IfIndex`	The interface on which the next hop gateway can be reached.
`Nexthop Name`	The name of next hop interface.
`Metric`	The metric of the route towards the destination.
`Preference`	The preference of the route towards destination.
`Refcnt`	Only used for debugging.

show ipv6 pim sparse-mode rp-hash

Use this command to display the Rendezvous Point (RP) to be chosen based on the IPv6 group address selected.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode rp-hash <IPv6-group-addr>`

Parameter	Description
<IPv6-group-addr>	The IPv6 group address used to find the RP, entered in the form X:X::X:X.

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 pim sparse-mode rp-hash ff04:10
```

Figure 53-12: Output from the show ipv6 pim sparse-mode rp-hash command:

```
awplus#show ipv6 pim sparse-mode rp-hash ff04::10
    RP: 3ffe:10:10:5::153
    Info source: 3ffe:10:10:5::153, via bootstrap
```

Related commands **show ipv6 pim sparse-mode rp mapping**

show ipv6 pim sparse-mode rp mapping

Use this command to show group-to-RP (Rendezvous Point) mappings, and the RP set.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode rp mapping`

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 pim sparse-mode rp mapping
```

Figure 53-13: Output from the show ipv6 pim sparse-mode rp mapping command

```
awplus#show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
Group(s): ff00::/8
  RP: 3ffe:10:10:5::153
    Info source: 3ffe:10:10:5::153, via bootstrap, priority 192
         Uptime: 05:36:40
```

Related commands **show ipv6 pim sparse-mode rp-hash**

show ipv6 pim sparse-mode rp nexthop

Use this command to display the RP (Rendezvous Point) next hop information used by PIM-SMv6.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 pim sparse-mode rp nexthop <RP-group-addr>`

Parameter	Description
<RP-group-addr>	Specify the RP group address used to display next hop RP information, entered in the form X:X::X:X.

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 pim sparse-mode rp nexthop 3ffe:10:10:5::153
```

Figure 53-14: Example output from the show ipv6 pim sparse-mode rp nexthop command

```
awplus#show ipv6 pim sparse-mode rp nexthop 3ffe:10:10:5::153
Flags: N = New, R = RP, S = Source, U = Unreachable
Destination           Type  Nexthop Nexthop Nexthop  Nexthop Metric     Pref    Refcnt
                            Num     Addr    Ifindex  Name
-------------------------------------------------------------------------------------
3ffe:10:10:5::153     .RS.  1       fe80::20e:cff:fe01:facc  2     30   110     1
```

Table 53-3: Parameters in output of the show ipv6 pim sparse-mode rp nexthop command

Parameter	Description
`Destination`	The destination address for which PIM-SMv6 requires next hop information.
`Type`	The type of destination, as indicated by the Flags description. N = New, R= RP, S = Source, U = Unreachable.
`Nexthop Num`	The number of next hops to the destination. PIM-SMv6 always uses only 1 next hop.
`Nexthop Addr`	The address of the primary next hop gateway.
`Nexthop IfIndex`	The interface on which the next hop gateway can be reached.
`Nexthop Name`	The name of next hop interface.
`Metric`	The metric of the route towards the destination.
`Preference`	The preference of the route towards destination.
`Refcnt`	Only used for debugging.

undebug all ipv6 pim sparse-mode

Use this command to disable all PIM-SMv6 debugging.

Syntax `undebug all ipv6 pim sparse-mode`

Mode Privileged Exec

Example

```
awplus# undebug all ipv6 pim sparse-mode
```

Related commands **debug ipv6 pim sparse-mode**

undebug ipv6 pim sparse-mode

Use this command to deactivate PIM-SMv6 debugging. Note that this command is an alias of the **no** variant of the **debug ipv6 pim sparse-mode** command.

Syntax

```
undebug ipv6 pim sparse-mode [all] [events] [mfc] [mib] [nexthop]
    [nsm] [state] [timer]
```

Parameter	Description
`all`	Deactivates all PIM-SMv6 debugging.
`events`	Deactivates debug printing of PIM-SMv6 events.
`mfc`	Deactivates debug printing of MFC (Multicast Forwarding Cache).
`mib`	Deactivates debug printing of PIM-SMv6 MIBs.
`nexthop`	Deactivates debug printing of PIM-SMv6 next hop communications.
`nsm`	Deactivates debugging of PIM-SMv6 NSM (Network Services Module) communications.
`state`	Deactivates debug printing of state transition on all PIM-SMv6 FSMs.
`timer`	Deactivates debug printing of PIM-SMv6 timers.

Mode Privileged Exec and Global Configuration

Example

```
        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# undebug ipv6 pim sparse-mode all

        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# undebug ipv6 pim sparse-mode events

        awplus# configure terminal
awplus(config)# terminal monitor
awplus(config)# undebug ipv6 pim sparse-mode nexthop
```

Validation Output

Figure 53-15: Example output from the show debugging ipv6 pim sparse-mode command after issuing the undebug ipv6 pim sparse-mode all command

```
awplus#undebug ipv6 pim sparse-mode all
awplus#show debugging ipv6 pim sparse-mode
PIM-SMv6 debugging status:
  PIM event debugging is off
  PIM MFC debugging is off
  PIM state debugging is off
  PIM packet debugging is off
  PIM Hello HT timer debugging is off
  PIM Hello NLT timer debugging is off
  PIM Hello THT timer debugging is off
  PIM Join/Prune JT timer debugging is off
  PIM Join/Prune ET timer debugging is off
  PIM Join/Prune PPT timer debugging is off
  PIM Join/Prune KAT timer debugging is off
  PIM Join/Prune OT timer debugging is off
  PIM Assert AT timer debugging is off
  PIM Register RST timer debugging is off
  PIM Bootstrap BST timer debugging is off
  PIM Bootstrap CRP timer debugging is off
  PIM mib debugging is off
  PIM nsm debugging is off
  PIM nexthop debugging is off
```

Related commands

debug ipv6 pim sparse-mode
show debugging ipv6 pim sparse-mode
undebug all ipv6 pim sparse-mode

Chapter 54: PIM-DM Introduction and Configuration

Introduction 54.2
Characteristics of PIM-DM 54.2
PIM-DM Terminology 54.3
PIM-DM Configuration 54.4
    Configuration Example 54.4
    Verifying Configuration 54.6

Introduction

Protocol Independent Multicast - Dense Mode (PIM-DM) is a data-driven multicast routing protocol, which builds source-based multicast distribution trees that operate on the Flood-and-Prune principle. It requires unicast-reachability information, but does not depend on a specific unicast routing protocol.

For details of the commands used to configure PIM-DM, see **Chapter 55, PIM-DM Commands**. For a general overview of multicasting, see **Chapter 47, Multicast Introduction and Commands**.

Characteristics of PIM-DM

PIM Dense Mode (PIM-DM) is a significantly less complex protocol than PIM Sparse Mode (PIM-SM). PIM-DM works on the principle that it is probable that any given multicast stream will have at least one downstream listener. PIM-DM is ideal where many hosts subscribe to receive multicast packets, so most of the PIM Routers receive and forward all multicast packets.

Where PIM-SM only forwards a multicast stream when requested, PIM-DM always floods any new multicast stream that arrives at the PIM Router and only stops flooding the multicast stream on a given link if it is explicitly told to, by receiving a Prune message from the downstream PIM Router.

PIM-DM does not include the concepts of Rendezvous Points, which are used in PIM-SM. PIM-SM explicitly builds unidirectional shared trees rooted at a Rendezvous Point (RP) per group. PIM-DM implicitly builds shortest-path trees by flooding multicast traffic domain wide, then Prunes back branches of the tree where no receivers are available. As with PIM-SM, so does PIM-DM also use Reverse Path Forwarding (RPF) to stop loops for packet forwarding for PIM Routers receiving multicast packets.

PIM-DM Terminology

See the below descriptions of the terms and concepts used to describe the PIM-DM protocol:

PIM router Any Layer 3 routing device that is running PIM, such as an Allied Telesis managed Layer 3 switch or Allied Telesis router.

Reverse path forwarding Reverse Path Forwarding (RPF) is the mechanism that PIM uses to make sure it does not forward multicast streams around in loops. If a set of PIM Routers are connected in a loop, and each PIM Router is forwarding a given multicast stream, then eventually the multicast stream would be forwarded right around the loop.

To prevent this from happening, PIM makes use of the fact that the unicast routing tables in a set of PIM Routers should converge into a loop-free tree of paths to any given destination.

When a PIM Router receives a multicast stream from source address `SourceA` through an interface `IF1`, it checks whether `IF1` is the interface the PIM Router would use to reach `SourceA`. The PIM Router will only forward the multicast stream if `IF1` is the interface the PIM Router would use to reach `SourceA`.

RPF determines whether the interface is correct by consulting unicast routing tables. This ensure that the multicast stream is forwarded in a loop-free manner back up the tree of unicast paths that lead to the source.

Forwarding multicast packets PIM Routers forward a given multicast stream onto all PIM enabled IP interfaces that have not received a Prune for the given multicast stream. As with unicast routing, the PIM Router decrements the TTL (Time To Live) in each packet that the PIM Router forwards. The packet is discarded if the TTL is decremented to 0.

However, unlike unicast routing, the destination MAC addresses of the packets are not altered as they are forwarded by the PIM Router. The destination MAC addresses remain set to the multicast MAC addresses that correspond to the destination group address of the multicast stream.

Upstream Towards the Source.

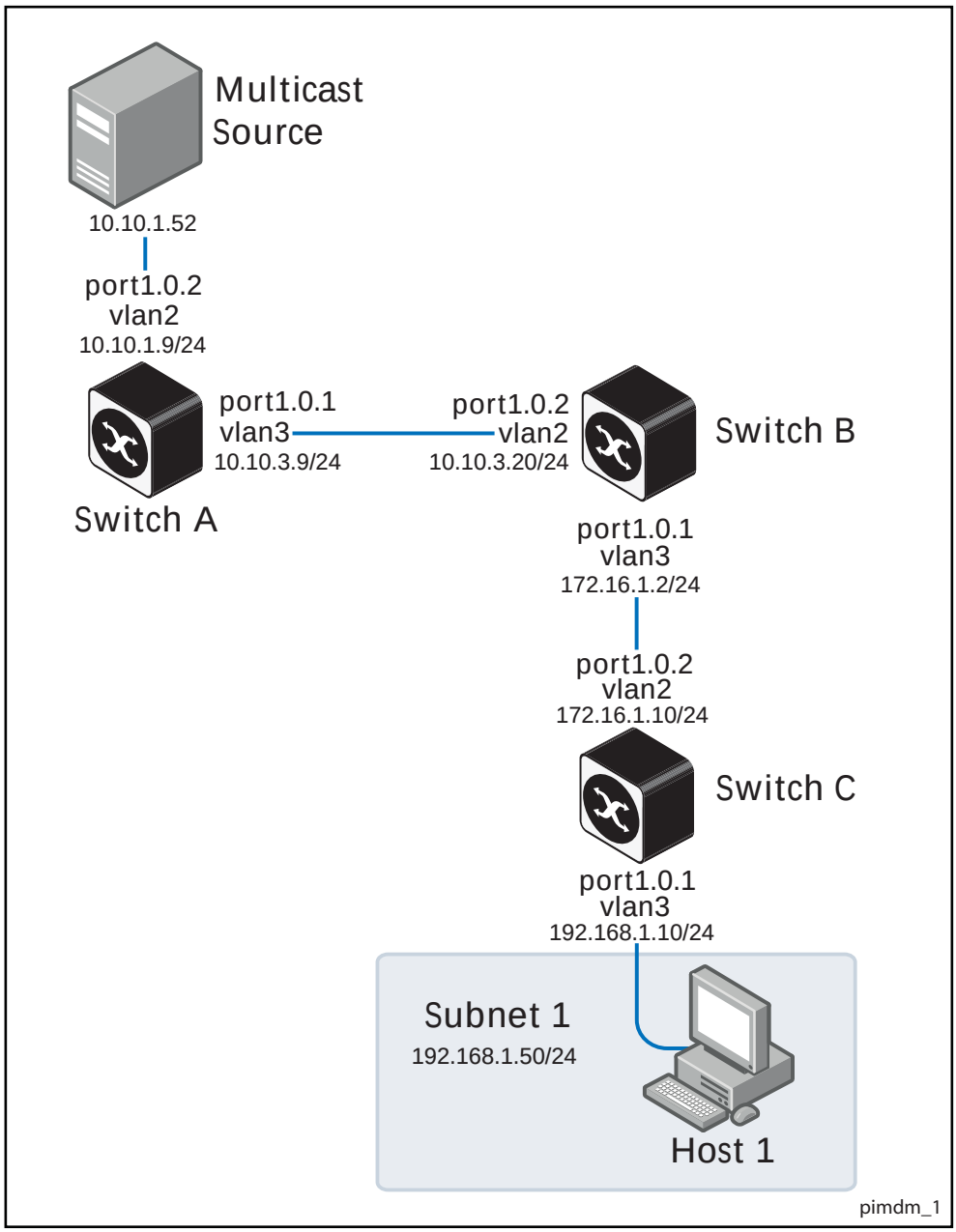
Downstream Anything other than the upstream interface for that group.

PIM-DM Configuration

The main requirement is to enable PIM-DM on the desired interfaces. This section provides a PIM-DM configuration example for a relevant scenario. The configuration uses Allied Telesis managed Layer 3 Switches as the PIM Routers. Three PIM Routers are connected in a chain, and a multicast client is attached to the third PIM Router.

Configuration Example

In this example, the address of the multicast source is 10.10.1.52. The following figure displays the network topology used in this example:



The steps involved in the forwarding of the multicast streams for this sample configuration are:

Switch A

1. When the PIM Routers start, they use the exchange of PIM Hello packets for PIM neighbor relationships with each other. Then each PIM Router becomes aware of the location of its PIM neighbors.

2. As a multicast stream arrives from the source to **Switch A**, it performs an RPF check on the source IP address of the multicast stream. **Switch A** determines the best route to the source IP address (10.10.1.52) is the receiving interface, so it forwards the multicast stream to its only PIM neighbor.

3. **Switch A** creates an (S, G) (Source, Group) entry in its PIM-DM forwarding table. Any further packets from the same source, which are destined to be forwarded to the same group, will be automatically forwarded without an RFP (Reverse Path Forwarding) check.

Switch B

4. When the multicast stream arrives at **Switch B**, it performs the same steps (2 and 3) as **Switch A**. This results in **Switch B** also having an (S, G) entry for the multicast stream in its PIM forwarding table, and the multicast stream is forwarded to **Switch C**.

Switch C

5. When the multicast stream arrives at **Switch C**, it will perform an RPF check on the multicast stream as it arrives, and accept it.

   This PIM Router does not have any downstream PIM Routers, but if **Switch C** has received an IGMP report from the client to request this multicast stream, **Switch C** will forward the multicast stream out port1.0.1, but no other ports.

   If the client leaves the group, and **Switch C** has no other attached clients requesting the group, then **Switch C** will send a Prune message upstream, resulting in **Switch A** and **Switch B** stopping forwarding the multicast stream to **Switch C**.

Switch A configuration output

See the following configuration output for **Switch A**:

```
hostname Switch A
vlan database
vlan 2 state enable
vlan 3 state enable
interface vlan2
ip address 10.10.1.9/24
ip igmp
ip pim dense-mode
!
interface vlan3
ip address 10.10.3.9/24
ip igmp
ip pim dense-mode
!
interface port1.0.1
switchport access vlan 3
!
interface port1.0.2
switchport access vlan 2
!
ip multicast-routing
!
```

Switch B configuration output

See the following configuration output for **Switch B**:

```
hostname Switch B
vlan database
vlan 2 state enable
vlan 3 state enable
interface vlan2
ip address 10.10.3.20/24
ip igmp
ip pim dense-mode
!
interface vlan3
ip address 172.16.1.2/24
ip igmp
ip pim dense-mode
!
interface port1.0.1
switchport access vlan 3
!
interface port1.0.2
switchport access vlan 2
!
ip multicast-routing
!
```

Switch C configuration output

See the following configuration output for **Switch C**:

```
hostname Switch C
vlan database
vlan 2 state enable
vlan 3 state enable
interface vlan2
ip address 172.16.1.10/24
ip igmp
ip pim dense-mode
!
interface vlan3
ip address 192.168.1.10/24
ip igmp
ip pim dense-mode
!
interface port1.0.1
switchport access vlan 3
!
interface port1.0.2
switchport access vlan 2
!
ip multicast-routing
!
```

Verifying Configuration

Use the following commands to verify the interface details and multicast routing table.

Interface details

The **show ip pim dense-mode interface** command displays the interface details for **Switch C**.

```
Address          Interface VIFindex Ver/   Nbr
                                    Mode   Count
192.168.1.10     port1.0.1  0       v2/D   0
172.16.1.10      port1.0.2  2       v2/D   1
```

IP multicast routing table

The **show ip mroute** command displays the IP multicast routing table (for **Switch C**).

```
IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder
installed
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)

(10.10.1.52, 224.0.1.3), uptime 00:00:15
Owner PIM-DM, Flags: F
  Incoming interface: port1.0.2
  Outgoing interface list:
    port1.0.1 (1)
```

IP PIM-DM multicast routing table

The **show ip pim dense-mode mroute** command displays the IP PIM-DM multicast routing table (for **Switch C**).

```
PIM-DM Multicast Routing Table
(10.10.1.52, 224.0.1.3)
  RPF Neighbor: 172.16.1.2, Nexthop: 172.16.1.2, port1.0.2
  Upstream IF: port1.0.2
    Upstream State: Forwarding
    Assert State: NoInfo
  Downstream IF List:
    port1.0.1, in 'olist':
      Downstream State: NoInfo
      Assert State: NoInfo
```

Chapter 55: PIM-DM Commands

Command List 55.2
debug pim dense-mode all 55.2
debug pim dense-mode context 55.3
debug pim dense-mode decode 55.4
debug pim dense-mode encode 55.4
debug pim dense-mode fsm 55.5
debug pim dense-mode mrt 55.6
debug pim dense-mode nexthop 55.7
debug pim dense-mode nsm 55.8
debug pim dense-mode vif 55.9
ip pim dense-mode 55.10
ip pim dense-mode passive 55.10
ip pim ext-srcs-directly-connected (PIM-DM) 55.11
ip pim hello-holdtime (PIM-DM) 55.11
ip pim hello-interval (PIM-DM) 55.12
ip pim max-graft-retries 55.13
ip pim neighbor-filter (PIM-DM) 55.15
ip pim propagation-delay 55.16
ip pim state-refresh origination-interval 55.17
show debugging pim dense-mode 55.18
show ip pim dense-mode interface 55.19
show ip pim dense-mode interface detail 55.20
show ip pim dense-mode mroute 55.21
show ip pim dense-mode neighbor 55.22
show ip pim dense-mode neighbor detail 55.23
show ip pim dense-mode nexthop 55.24
undebug all pim dense-mode 55.25

Command List

This chapter provides an alphabetical reference of PIM-DM commands. For commands common to PIM-SM and PIM-DM, see **Chapter 47, Multicast Introduction and Commands**.

debug pim dense-mode all

This command enables PIM-DM debugging.

The **no** variant of this command disables PIM-DM debugging.

Syntax

```
debug pim dense-mode all
no debug pim dense-mode all
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode all
```

Output **Figure 55-1: Example output from the debug pim dense-mode all command**

```
PIM event debugging is on
PIM MFC debugging is on
PIM state debugging is on
PIM packet debugging is on
PIM incoming packet debugging is on
PIM outgoing packet debugging is on
```

Validation Commands **show debugging pim dense-mode**

Related Commands
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode context

This command enables debugging of general configuration context.

The **no** variant of this command disables debugging of general configuration context.

Syntax

```
debug pim dense-mode context
no debug pim dense-mode context
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode context
```

Related Commands

debug pim dense-mode all
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode decode

This command enables debugging of the PIM-DM message decoder.

The **no** variant of this command disables debugging of the PIM-DM message decoder.

Syntax

```
debug pim dense-mode decode
no debug pim dense-mode decode
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode decoder
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode encode

This command enables debugging of the PIM-DM message encoder.

The **no** variant of this command disables debugging of the PIM-DM message encoder.

Syntax

```
debug pim dense-mode encode
no debug pim dense-mode encode
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode encoder
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode fsm

This command enables debugging of Finite-State Machine (FSM) specific information of all Multicast Routing Table (MRT) and MRT Virtual Multicast Interface (MRT-VIF) entries.

The **no** variant of this command disables debugging of Finite-State Machine (FSM) specific information of all Multicast Routing Table (MRT) and MRT Virtual Multicast Interface (MRT-VIF) entries.

Syntax

```
debug pim dense-mode fsm
no debug pim dense-mode fsm
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode fsm
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode mrt

This command enables debugging of MRT and MRT-VIF entry handling (for example, creation and deletion of).

The **no** variant of this command disables debugging of MRT and MRT-VIF entry handling.

Syntax

```
debug pim dense-mode mrt
no debug pim dense-mode mrt
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode mrt
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode nexthop

This command enables debugging of Reverse Path Forwarding (RPF) neighbor next hop cache handling.

The **no** variant of this command disables debugging of Reverse Path Forwarding (RPF) neighbor next hop cache handling.

Syntax

```
debug pim dense-mode nexthop
no debug pim dense-mode nexthop
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode nexthop
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nsm
debug pim dense-mode vif

debug pim dense-mode nsm

This command enables debugging of PIM-DM interface with NSM.

The **no** variant of this command disables debugging of PIM-DM interface with NSM.

Syntax

```
debug pim dense-mode nsm
no debug pim dense-mode nsm
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode nsm
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode vif

debug pim dense-mode vif

This command enables debugging of VIF handling.

The **no** variant of this command disables debugging of VIF handling.

Syntax

```
debug pim dense-mode vif
no debug pim dense-mode vif
```

Mode Privileged Exec and Global Configuration

Example

```
awplus# configure terminal
awplus(config)# debug pim dense-mode vif
```

Related Commands

debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm

ip pim dense-mode

This command enables or disables PIM-DM operation from Interface mode on the current VLAN interface. This command also disables passive mode on the VLAN interface if passive mode has been enabled using an **ip pim dense-mode passive** command.

The **no** variant of this command disables all PIM-DM activities on the interface.

Syntax

```
ip pim dense-mode
no ip pim dense-mode
```

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim dense-mode
```

ip pim dense-mode passive

This command enables PIM-DM passive mode operation from Interface mode on the current VLAN interface.

The **no** variant of this command disables passive mode.

Syntax

```
ip pim dense-mode passive
no ip pim dense-mode passive
```

Mode Interface Configuration for a VLAN interface.

Usage Configuring a VLAN interface as a passive PIM-DM interface indicates that the VLAN interface is connected to a stub network (i.e. a network that does not contain any PIM Routers). So, multicast streams that arrive on other PIM-DM interfaces can be routed to hosts on the passive PIM-DM interface, but no PIM neighbor relationships will be formed on the passive PIM-DM interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim dense-mode passive
```

ip pim ext-srcs-directly-connected (PIM-DM)

Use this command to configure PIM to treat all source traffic arriving on the interface as though it was sent from a host directly connected to the interface.

For a more detailed description of this command, see: **Chapter 51, PIM-SM Commands** - **ip pim ext-srcs-directly-connected (PIM-SM)** command on page 51.16.

ip pim hello-holdtime (PIM-DM)

This command configures a **hello-holdtime**. The PIM **hello-holdtime** on a VLAN interface is the period which the router will wait to receive a hello from neighbors on that interface. If the router does not receive a hello from a given neighbor within that period, then it will decide that the neighbor is no longer an active PIM Router, and will terminate the neighbor relationship.

You cannot configure a **hello-holdtime** value that is less than the current **hello-interval**. Each time the **hello-interval** is updated, the **hello-holdtime** is also updated, according to the following rules:

- If the **hello-holdtime** is not configured; or if the hello holdtime is configured and less than the current **hello-interval** value, it is modified to 3.5 times the **hello-interval** value.
- Otherwise, it retains the configured value.

Use the **no** variant of this command to return the hello-holdtime value to its default of 3.5 times the current hello-interval value.

Syntax

```
ip pim hello-holdtime <holdtime>
no ip pim hello-holdtime
```

Parameter	Description
<holdtime>	<1-65535> The holdtime value in seconds (no fractional seconds are accepted).

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim hello-holdtime 123
```

ip pim hello-interval (PIM-DM)

This command configures a PIM **hello-interval** value. The PIM **hello-interval** on a VLAN interface is the period at which the router will transmit PIM hello messages on that interface.

When the **hello-interval** is configured, and the **hello-holdtime** is not configured, or when the configured **hello-holdtime** value is less than the new **hello-interval** value; the **hello-holdtime** value is modified to 3.5 times the **hello-interval** value. Otherwise, the **hello-holdtime** value is the configured value. The default is 30 seconds.

Use the **no** variant of this command to reset the **hello-interval** to the default.

Syntax

```
ip pim hello-interval <interval>
no ip pim hello-interval
```

Parameter	Description
<interval>	*<1-65535>* The value in seconds (no fractional seconds accepted).

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim hello-interval 123
```

ip pim max-graft-retries

This command configures PIM-DM to send a limited number of Graft message retries, after which time the device will remove all information regarding the particular (Source, Group), or until the device receives an acknowledgment, whichever occurs first.

The **no** variant of this command configures PIM-DM to send Graft message retries until the device receives an acknowledgment, which is the default behavior.

Syntax

```
ip pim max-graft-retries <1-65535>
no pim max-graft-retries
```

Parameter	Description
`no`	Negate a command or set its defaults.
`ip`	Internet Protocol (IP).
`pim`	PIM Interface commands.
`max-graft-retries`	PIM Graft message retries.
<1-65535>	Graft message retries before ceasing Graft message retries.

Default By default, Graft retries are sent by PIM-DM until the device receives an acknowledgment.

Mode Interface Configuration for a VLAN interface.

Usage Graft messages are used to reduce the join latency when a previously pruned branch of the source tree must be grafted back, when a member joins the group after the PIM-DM device has sent a Prune message to prune unwanted traffic. Graft messages are the only PIM-DM messages that receive an acknowledgment.

If Graft messages were not used, then the member waiting for pruned off traffic would have to wait up to 3 minutes for the periodic re-flooding to occur to begin receiving multicast traffic again. By using Grafts, the Prune can be reversed much faster than waiting for periodic re-flooding to begin receiving multicast traffic again.

Examples To configure PIM-DM on the VLAN interface `vlan2` to send a maximum of 10 Graft message retries, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim max-graft-retries 10
```

To configure PIM-DM on the VLAN interface `vlan2` to send Graft message retries forever, which is the default behavior, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim max-graft-retries
```

Validation Commands

show ip mroute
show ip pim dense-mode mroute
show running-config

ip pim neighbor-filter (PIM-DM)

Enables filtering of neighbors on the VLAN interface. When configuring a neighbor filter, PIM-DM will either not establish adjacency with the neighbor, or terminate adjacency with the existing neighbors if denied by the filtering access list.

Use the **no** variant of this command to disable this function.

Syntax

```
ip pim neighbor-filter [<number>|<accesslist>]
no ip pim neighbor-filter [<number>|<accesslist>]
```

Parameter	Description
<number>	*<1-99>* Standard IP access list number.
<accesslist>	IP access list name.

Default By default, there is no filtering.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim neighbor-filter 14
```

ip pim propagation-delay

This command configures the PIM **propagation-delay** value. The PIM **propagation-delay** is the expected delay in the transfer of PIM messages across the VLAN interface that it is attached to.

Use the **no** variant of this command to return the **propagation-delay** to the default (1000 milliseconds).

Syntax

```
ip pim propagation-delay <delay>
no ip pim propagation-delay
```

Parameter	Description
<delay>	*<1000-5000>* The value in milliseconds. The default is 1000 milliseconds.

Default The propagation-delay is set to 1000 milliseconds by default.

Mode Interface Configuration for a VLAN interface.

Examples

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim propagation-delay 2000

awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip pim propagation-delay
```

ip pim state-refresh origination-interval

This command configures a PIM **state-refresh origination-interval** value. The origination interval is the number of seconds between PIM state refresh control messages. The default is 60 seconds.

Use the **no** variant of this command to return the origination interval to the default.

Syntax

```
ip pim state-refresh origination-interval <interval>
no ip pim state-refresh origination-interval
```

Parameter	Description
<interval>	*<1-100>* The integer value in seconds (no fractional seconds accepted). The default **state-refresh origination-interval** value is 60.

Default The state-refresh origination-interval is set to 60 seconds by default, and is reset using negation.

Mode Interface Configuration for a VLAN interface.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ip pim state-refresh origination-interval 65
```

show debugging pim dense-mode

This command displays the status of the debugging of the system.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging pim dense-mode`

Mode User Exec and Privileged Exec

Output **Figure 55-2: Example output from the show debugging pim dense-mode command**

```
PIM-DM Debugging status:
  PIM-DM Decoder debugging is off
  PIM-DM Encoder debugging is off
  PIM-DM FSM debugging is off
  PIM-DM MRT debugging is off
  PIM-DM NHOP debugging is off
  PIM-DM NSM debugging is off
  PIM-DM VIF debugging is off
```

Related Commands **debug pim dense-mode all**

show ip pim dense-mode interface

This command displays the PIM-DM interface information.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show ip pim dense-mode interface`

Mode User Exec and Privileged Exec

Example To display information about the PIM-DM interfaces, use the command:

```
awplus# show ip pim dense-mode interface
```

Output

```
Total configured interfaces: 24   Maximum allowed: 32
Total active interfaces:     22

Address          Interface VIFIndex Ver/   Nbr
                                    Mode   Count
192.168.1.53/24  vlan2     0        v2/D   2
192.168.2.1      vlan3     2        v2/D   0
...
Note that this screen has been edited to remove any additional
interfaces.
```

Table 55-1: Parameters in the output of the show ip pim dense-mode interface command

Parameter	Description
`Total configured interfaces`	The number of configured PIM Dense Mode interfaces.
`Maximum allowed`	The maximum number of PIM Dense Mode interfaces that can be configured.
`Total active interfaces`	The number of active PIM Dense Mode interfaces.
`Address`	Primary PIM-DM address.
`Interface`	Name of the PIM-DM interface.
`VIF Index`	The Virtual Interface index of the VLAN.
`Ver/Mode`	PIM version/Dense mode.
`Nbr Count`	Neighbor count of the PIM-DM interface.

Related Commands ip pim dense-mode
show ip pim dense-mode neighbor

show ip pim dense-mode interface detail

This command displays detailed information on a PIM-DM interface.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip pim dense-mode interface detail`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip pim dense-mode interface detail
```

Output **Figure 55-3: Example output from the show ip pim dense-mode interface detail command**

```
vlan2 (vif-id: 0):
  Address 192.168.1.53/24
  Hello period 30 seconds, Next Hello in 30 seconds
  Neighbors:
    192.168.1.152/32
    192.168.1.149/32
vlan3 (vif-id: 2):
  Address 192.168.10.53/24
  Hello period 30 seconds, Next Hello in 8 seconds
  Neighbors: none
```

show ip pim dense-mode mroute

This command displays the IP PIM-DM multicast routing table.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip pim dense-mode mroute`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip pim dense-mode mroute
```

Output **Figure 55-4: Example output from the show ip pim dense-mode mroute command**

```
PIM-DM Multicast Routing Table
(192.168.10.52, 224.1.1.1)
  Source directly connected on vlan3
  State-Refresh Originator State: Originator
  Upstream IF: vlan3, State: Forwarding
  Downstream IF List:
    vlan2, in 'olist':
      Downstream State: NoInfo
      Assert State: NoInfo
```

show ip pim dense-mode neighbor

This command displays PIM-DM neighbor information.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip pim dense-mode neighbor`

Mode User Exec and Privileged Exec

Usage The total number of PIM-DM neighbors is restricted to 500 PIM-DM neighbors.

When the 500 PIM-DM neighbor limit is reached, as a result of receiving hello packets from new PIM-DM neighbors, a log entry will be issued to the log file in the below format:

```
<date> <time> <facility>.<severity> <program[<pid>]>: <message>
---------------------------------------------------------------
2008 Dec 10 00:58:39 user.err x900 PIM-DM[1150]: [VIF] Nbr
Create: Cannot create more than 500 neighbours - ignoring
neighbour 100.0.1.247/32 on vlan100
```

Example

```
awplus# show ip pim dense-mode neighbor
```

Output **Figure 55-5: Example output from the show ip pim dense-mode neighbor command**

```
Total number of neighbors: 500
Neighbor-Address  Interface          Uptime/Expires    Ver
192.168.1.152     vlan2                17:15:42/00:01:28 v2
192.168.1.149     vlan2                17:15:34/00:01:34 v2
```

show ip pim dense-mode neighbor detail

This command displays detailed PIM-DM neighbor information.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip pim dense-mode neighbor detail
```

Mode

User Exec and Privileged Exec

Example

```
awplus# show ip pim dense-mode neighbor detail
```

Output

Figure 55-6: Example output from the show ip pim dense-mode neighbor detail command

```
Neighbor 192.168.1.152 (vlan2)
 Up since 17:16:20, Expires in 00:01:20
Neighbor 192.168.1.149 (vlan2)
 Up since 17:16:12, Expires in 00:01:26
```

show ip pim dense-mode nexthop

This command displays the next hop information as used by PIM-DM. In the context of PIM-DM, the term '**next hop**' refers to the next hop router on the path back to the source address of a multicast stream.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip pim dense-mode nexthop`

Mode User Exec and Privileged Exec

Example

```
awplus# show ip pim dense-mode nexthop
```

Output **Figure 55-7: Example output from the show ip pim dense-mode neighbor nexthop command**

```
Destination      Nexthop    Nexthop     Nexthop        Metric Pref
                 Num        Addr        Interface
192.168.10.52    1          0.0.0.0     vlan2          3      1
```

Table 55-2: Parameters in the output of the show ip pim dense-mode neighbor nexthop command

Parameter	Description
`Destination`	Destination address for which PIM-DM requires next hop information.
`Nexthop Num`	Number of next hops to the destination. PIM can only use one next hop.
`Nexthop Addr`	Address of the current next hop gateway.
`Nexthop Interface`	Name of the next hop interface.
`Metric`	Metric of the route towards the destination.
`Preference`	Preference of the route towards the destination.

undebug all pim dense-mode

Use this command from the Global Configuration mode to disable all PIM-DM debugging.

Syntax `undebug all pim dense-mode`

Mode Global Configuration

Example

```
awplus# configure terminal
awplus(config)# undebug all pim dense-mode
```

Related Commands
debug pim dense-mode all
debug pim dense-mode context
debug pim dense-mode decode
debug pim dense-mode encode
debug pim dense-mode fsm
debug pim dense-mode mrt
debug pim dense-mode nexthop
debug pim dense-mode nsm
debug pim dense-mode vif

Chapter 56: MLD and MLD Snooping Introduction and Commands

MLD Introduction 56.2
MLD Snooping 56.3
MLD Snooping Configuration Examples 56.4
Command List 56.7
clear ipv6 mld 56.7
clear ipv6 mld group 56.8
clear ipv6 mld interface 56.8
debug mld 56.9
ipv6 mld 56.11
ipv6 mld access-group 56.12
ipv6 mld immediate-leave 56.13
ipv6 mld last-member-query-count 56.14
ipv6 mld last-member-query-interval 56.15
ipv6 mld limit 56.16
ipv6 mld querier-timeout 56.18
ipv6 mld query-interval 56.19
ipv6 mld query-max-response-time 56.20
ipv6 mld robustness-variable 56.21
ipv6 mld snooping 56.22
ipv6 mld snooping fast-leave 56.24
ipv6 mld snooping mrouter 56.25
ipv6 mld snooping querier 56.27
ipv6 mld snooping report-suppression 56.28
ipv6 mld static-group 56.30
ipv6 mld version 56.33
show debugging mld 56.34
show ipv6 mld groups 56.35
show ipv6 mld interface 56.36
show ipv6 mld snooping mrouter 56.37
show ipv6 mld snooping statistics 56.38

MLD Introduction

Multicast Listener Discovery (MLD) is used to exchange membership status information between IPv6 routers that support multicasting and members of multicast groups on a network segment. Host membership in a multicast group is reported by individual member hosts, and membership status is periodically polled by multicast routers. For a general overview of multicasting, see **Chapter 47, Multicast Introduction and Commands**.

MLD is defined in RFC 2710, "Multicast Listener Discovery (MLD) for IPv6". MLDv2 is defined in RFC 3810, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6".

AlliedWare Plus supports both RFC 2710 for MLD and RFC 3810 for MLDv2.

Note There is a 100 MLD interface limit when applying MLD commands to multiple VLANs. Only the first 100 VLANs have the required multicast structures added to the interfaces that allow multicast routing. x510 series switches have a 128 MLD group limit for (*, G) and (S,G) entries. See the limits for MLD interfaces depending on the number of VLANs, ports, static and dynamic groups as shown in the relevant product data sheet for your switch.

MLD requires memory for storing data structures, as well as the hardware tables to implement hardware routing. As the number of ports, VLANs, static and dynamic groups increases then more memory is consumed. You can track the memory used for MLD with the command:

```
awplus# show memory pools nsm | grep MLD
```

Static and dynamic groups (LACP), ports and VLANs are not limited for MLD. For VLANs, this allows you to configure MLD across more VLANs with fewer ports per VLAN, or fewer VLANs with more ports per VLAN. For LACPs, you can configure MLD across more LACP groups with fewer ports per LACP, or fewer LACP groups with more ports per LACP.

Sample running-config showing MLD configuration on interface vlan2

```
!
ipv6 forwarding
!
ipv6 multicast-routing
!
interface vlan2
  ipv6 address 2001:0db8::12:252/64
  ipv6 enable
  ipv6 mld
!
```

Note IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Note The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

Note The IPv6 Multicast addresses shown can be derived from IPv6 unicast prefixes as per RFC 3306.The IPv6 unicast prefix reserved for documentation is 2001:0db8::/32 as per RFC 3849. Using the base /32 prefix the IPv6 multicast prefix for 2001:0db8::/32 is ff3x:20:2001:0db8::/64. Where an RP address is 2001:0db8::1 the embedded RP multicast prefix is ff7x:120:2001:0db8::/96. For ASM (Any-Source Multicast) the IPV6 multicast addresses allocated for documentation purposes are ff0x::0db8:0:0/96 as per RFC 6676. This is a /96 prefix so that it can be used with group IDs as per RFC 3307. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

MLD Snooping

MLD Snooping is a feature whereby a Layer 2 switch listens to or "snoops" the MLD messages passing through the switch or from member hosts and multicast routers. The purpose of MLD Snooping is to provide efficient Layer 2 multicast forwarding, by sending only to hosts that have expressed an interest in receiving the multicast data.

Hosts express an interest in receiving multicast data for a given multicast group by sending an MLD join message. Without MLD Snooping, if one host expresses an interest in getting multicast data for a given group, by sending an MLD join for the multicast group, then all hosts connected to the same VLAN will also receive the multicast data. This wastes bandwidth on the switch ports connected to the host that are not interested in receiving the multicast data. Snooping takes note of exactly which ports have received joins for a given group, and send that group only to those ports.

MLD Snooping is enabled by default globally for the switch. It can be enabled and disabled on a per-VLAN basis.

MLD Snooping makes a distinction between Member ports, which are ports connected to members hosts, and Router ports, which are ports connected to, or directed towards, a Layer 3 router or a Layer 3 switch.

Figure 56-1: Sample running-config showing an MLD Snooping Querier set on vlan2

```
!
ipv6 mld snooping
!
interface vlan2
   ipv6 mld snooping querier
!
```

MLD Snooping Configuration Examples

For detailed information about the commands used to configure MLD Snooping, see **Chapter 56, MLD and MLD Snooping Introduction and Commands**.

The following examples configure MLD Snooping:

- **Enabling the MLD Snooping Querier on an interface**
- **Enabling MLD Snooping globally and on an interface**
- **Configuring a Multicast Router statically on an interface**
- **Enabling MLD Snooping Fast-Leave Processing on an interface**
- **Configuring MLD Snooping Report Suppression on an interface**

Enabling the MLD Snooping Querier on an interface

Use the MLD Snooping Querier to support MLD Snooping in a VLAN where PIM-SMv6 and MLD are not configured and whenever you do not need to route IPv6 multicast traffic.

You can configure the switch to generate MLD queries on a VLAN interface if multicast routing is not enabled. For each VLAN that is connected to switches that use MLD Snooping to report multicast traffic, configure one switch as the MLD Snooping Querier.

To enable and show MLD Snooping Querier on VLAN interface `vlan2`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping querier
awplus(config-if)# exit
awplus(config)# exit
awplus# show ipv6 mld interface vlan2
```

Note that the MLD Snooping Querier is configured in Interface Configuration mode only. You cannot configure MLD Snooping Querier globally for all VLAN interfaces on a switch.

Enabling MLD Snooping globally and on an interface

To globally enable and show MLD Snooping on the switch, enter the commands:

```
awplus# configure terminal
awplus(config)# ipv6 mld snooping
awplus(config)# exit
awplus# show ipv6 mld interface
```

Note that entering `show ipv6 mld interface` without an optional interface parameter displays MLD information for all configured interfaces globally on the switch.

To enable and show MLD Snooping on VLAN interface `vlan2`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping
awplus(config-if)# exit
awplus(config)# exit
awplus# show ipv6 mld interface vlan2
```

Note that entering `show ipv6 mld interface vlan2` with the optional interface parameter displays MLD information for that specified configured interface on the switch.

Configuring a Multicast Router statically on an interface

To configure and show a static connection to a Multicast Router for VLAN interface `vlan2`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping mrouter interface
                   port1.0.2
awplus(config-if)# exit
awplus(config)# exit
awplus# show ipv6 mld interface vlan2
```

Note the VLAN interface to the Multicast Router must be administratively up and the line protocol must be up to configure a static connection to a Multicast Router on the VLAN.

Enabling MLD Snooping Fast-Leave Processing on an interface

To enable and show MLD Snooping Fast-Leave Processing on VLAN interface `vlan2`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping fast-leave
awplus(config-if)# exit
awplus(config)# exit
awplus# show ipv6 mld interface vlan2
```

Configuring MLD Snooping Report Suppression on an interface

To enable and show MLD Snooping Report Suppression on VLAN interface `vlan2`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping report-suppression
awplus(config-if)# exit
awplus(config)# exit
awplus# show ipv6 mld interface vlan2
```

Command List

This chapter provides an alphabetical reference of configuration, clear, and show commands related to MLD and MLD Snooping.

The Multicast Listener Discovery (MLD) module includes the MLD Proxy service and MLD Snooping functionality. Some of the following commands may have commonalities and restrictions; these are described under the Usage section for each command.

Note IPv6 must be enabled on an interface with the **ipv6 enable** command, IPv6 forwarding must be enabled globally for routing IPv6 with the **ipv6 forwarding** command, and IPv6 multicasting must be enabled globally with the **ipv6 multicast-routing** command before using PIM-SMv6 commands.

Note The IPv6 Multicast addresses shown can be derived from IPv6 unicast prefixes as per RFC 3306.The IPv6 unicast prefix reserved for documentation is 2001:0db8::/32 as per RFC 3849. Using the base /32 prefix the IPv6 multicast prefix for 2001:0db8::/32 is ff3x:20:2001:0db8::/64. Where an RP address is 2001:0db8::1 the embedded RP multicast prefix is ff7x:120:2001:0db8::/96. For ASM (Any-Source Multicast) the IPV6 multicast addresses allocated for documentation purposes are ff0x::0db8:0:0/96 as per RFC 6676. This is a /96 prefix so that it can be used with group IDs as per RFC 3307. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

Note The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

clear ipv6 mld

Use this command to clear all MLD local memberships on all interfaces.

Syntax `clear ipv6 mld`

Mode Privileged Exec

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols and learned by MLD Snooping.

Example

```
awplus# clear ipv6 mld
```

Related Commands **clear ipv6 mld group**
clear ipv6 mld interface

clear ipv6 mld group

Use this command to clear MLD specific local-membership(s) on all interfaces, for a particular group.

Syntax `clear ipv6 mld group {*|<ipv6-address>}`

Parameter	Description
*	Clears all groups on all interfaces. This is an alias to the **clear ipv6 mld** command.
<ipv6-address>	Specify the group address for which MLD local-memberships are to be cleared from all interfaces. Specify the IPv6 multicast group address in the format in the format X:X::X:X.

Mode Privileged Exec

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols and learned by MLD Snooping.

Example

```
awplus# clear ipv6 mld group *
```

Related Commands
clear ipv6 mld
clear ipv6 mld interface

clear ipv6 mld interface

Use this command to clear MLD interface entries.

Syntax `clear ipv6 mld interface <interface>`

Parameter	Description
<interface>	Specifies name of the interface; all groups learned from this interface are deleted.

Mode Privileged Exec

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols and learned by MLD Snooping.

Example

```
awplus# clear ipv6 mld interface vlan2
```

Related Commands
clear ipv6 mld
clear ipv6 mld group

debug mld

Use this command to enable all MLD debugging modes, or a specific MLD debugging mode.

Use the **no** variant of this command to disable all MLD debugging modes, or a specific MLD debugging mode.

Syntax

```
debug mld {all|decode|encode|events|fsm|tib}
no debug mld {all|decode|encode|events|fsm|tib}
```

Parameter	Description
`all`	Debug all MLD.
`decode`	Debug MLD decoding.
`encode`	Debug MLD encoding.
`events`	Debug MLD events.
`fsm`	Debug MLD Finite State Machine (FSM).
`tib`	Debug MLD Tree Information Base (TIB).

Mode Privileged Exec and Global Configuration

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols and learned by MLD Snooping.

Examples

```
awplus# configure terminal
awplus(config)# debug mld all

awplus# configure terminal
awplus(config)# debug mld decode

awplus# configure terminal
awplus(config)# debug mld encode

awplus# configure terminal
awplus(config)# debug mld events
```

Output

```
Warning: Console logging enabled
awplus#05:15:00 awplus NSM[1406]: [MLD-DECODE] Dec V2 Grp Rec: Grp ff08::1 on
port2.0.1
05:15:00 awplus NSM[1406]: [MLD-DECODE] Dec V2 Grp Rec: G-Rec not found! on
port2.0.1 for ff08::1
05:15:00 awplus NSM[1406]: [MLD-FSM] Process Event: I=port2.0.1, G=ff08::1, State:
Include, Event: Change To Include
05:15:00 awplus NSM[1406]: [MLD-FSM] State Change: Include(1)->Include(1)
05:15:00 awplus NSM[1406]: [MLD-ENCODE] Send Grp - Src Report: HST-IF vlan1: No
Router Ports found
05:15:00 awplus NSM[1406]: [MLD-DECODE] Socket Read: Ignoring MLD Message on L3
socketsince Snooping is enabled on vlan1
05:15:01 awplus NSM[1406]: [MLD-DECODE] Dec V2 Grp Rec: Grp ff08::1 on port2.0.1
05:15:01 awplus NSM[1406]: [MLD-ENCODE] MLD Enc Hdr: MLD Listener Query Checksum
=8511, MsgLen=60
05:15:01 awplus NSM[1406]: [MLD-ENCODE] Send Group - Source Query: Sent G-S Query
on port2.0.1
05:15:01 awplus NSM[1406]: [MLD-FSM] State Change: Include(1)->Exclude(2)
05:15:01 awplus NSM[1406]: [MLD-TIB] Source Rec Del: S=2002::3 Intf=vlan1
05:15:01 awplus NSM[1406]: [MLD-ENCODE] Send Group Report: HST-IF vlan1: No Router
Ports found
05:15:01 awplus NSM[1406]: [MLD-DECODE] Socket Read: Ignoring MLD Message on L3
socketsince Snooping is enabled on vlan1
05:15:01 awplus NSM[1406]: [MLD-EVENTS] Grp - Src Report Rexmit: Exipry for Grp
ff08::1 on vlan1
05:15:01 awplus NSM[1406]: [MLD-EVENTS] Grp - Src Report Rexmit: Group-Source
Report Rexmit failed(-16)
05:15:02 awplus NSM[1406]: [MLD-EVENTS] Grp - Src Query Rexmit: Exipry for Grp
ff08::1 on port2.0.1
05:15:02 awplus NSM[1406]: [MLD-ENCODE] MLD Enc Hdr: MLD Listener Query
Checksum=8511, MsgLen=60
05:15:02 awplus NSM[1406]: [MLD-ENCODE] Send Group - Source Query: Sent G-S Query
on port2.0.1
05:15:02 awplus NSM[1406]: [MLD-EVENTS] Grp Report Rexmit: Exipry for Grp ff08::
1 on vlan1
05:15:02 awplus NSM[1406]: [MLD-ENCODE] Send Group Report: HST-IF vlan1: No Router
Ports found
05:15:02 awplus NSM[1406]: [MLD-EVENTS] Grp - Src Report Rexmit: Exipry for Grp
ff08::1 on vlan1
05:15:02 awplus NSM[1406]: [MLD-TIB] Source Rec Del: S=2002::3 Intf=vlan1
05:15:03 awplus NSM[1406]: [MLD-EVENTS] Src - Rec Liveness Timer: Exipry for Src
 2002::3 on port2.0.1
005:15:03 awplus NSM[1406]: [MLD-FSM] Process Event: I=port2.0.1, G=ff08::1,
State: Exclude, Event: Source Tmr Expry
05:15:03 awplus NSM[1406]: [MLD-FSM] State Change: Exclude(2)->Exclude(2)
05:15:03 awplus NSM[1406]: [MLD-FSM] Host Process Event: I=vlan1, G=ff08::1,
05:15:06 awplus appmond[1244]: monitoring imi memory usage (max:51200000 kB)
05:15:06 awplus appmond[1244]: monitoring rmond memory usage (max:51200000 kB)
05:15:06 awplus appmond[1244]: monitoring lldpd memory usage (max:51200000 kB)
05:15:06 awplus NSM[1406]: [MLD-EVENTS] Querier Timer: Exipry on port2.0.1, Send
ing General Query
05:15:06 awplus NSM[1406]: [MLD-ENCODE] MLD Enc Hdr: MLD Listener Query Checksum
=14706, MsgLen=28
05:15:06 awplus NSM[1406]: [MLD-ENCODE] Send Gen Query: Sent General Query on
port2.0.1, ret=90
05:15:06 awplus NSM[1406]: [MLD-EVENTS] Querier Timer: Exipry on port2.0.1,
Sending General Query
05:15:06 awplus NSM[1406]: [MLD-ENCODE] MLD Enc Hdr: MLD Listener Query Checksum
=14706, MsgLen=28
05:15:06 awplus NSM[1406]: [MLD-ENCODE] Send Gen Query: Sent General Query on
port2.0.1, ret=90
05:15:06 awplus NSM[1406]: [MLD-EVENTS] Querier Timer: Exipry on port2.0.1,
Sending General Query
05:15:06 awplus NSM[1406]: [MLD-ENCODE] MLD Enc Hdr: MLD Listener Query Checksum
=14706, MsgLen=28
05:15:06 awplus NSM[1406]: [MLD-ENCODE] Send Gen Query: Sent General Query on po
rt2.0.1, ret=90
```

Related Commands **show debugging mld**

ipv6 mld

Use this command to enable the MLD protocol operation on an interface. This command enables MLD protocol operation in stand-alone mode, and can be used to learn local-membership information prior to enabling a multicast routing protocol on the interface.

Use the **no** variant of this command to return all MLD related configuration to the default (including MLD Snooping).

Note

There is a 100 MLD interface limit when applying MLD commands to multiple VLANs. Only the first 100 VLANs have the required multicast structures added to the interfaces that allow multicast routing. x510 series switches have a 128 MLD group limit for (*, G) and (S,G) entries. See the limits for MLD interfaces depending on the number of VLANs, ports, static and dynamic groups as shown in the relevant product data sheet for your switch.

Syntax

```
ipv6 mld
no ipv6 mld
```

Default MLD is disabled by default.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage MLD requires memory for storing data structures, as well as the hardware tables to implement hardware routing. As the number of ports, VLANs, static and dynamic groups increases then more memory is consumed. You can track the memory used for MLD with the command:

```
awplus# show memory pools nsm | grep MLD
```

Static and dynamic groups (LACP), ports and VLANs are not limited for MLD. For VLANs, this allows you to configure MLD across more VLANs with fewer ports per VLAN, or fewer VLANs with more ports per VLAN. For LACPs, you can configure MLD across more LACP groups with fewer ports per LACP, or fewer LACP groups with more ports per LACP.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan1
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld
```

ipv6 mld access-group

Use this command to control the multicast local-membership groups learned on an interface.

Use the **no** variant of this command to disable this access control.

Syntax

```
ipv6 mld access-group <IPv6-access-list-name>
no ipv6 mld access-group
```

Parameter	Description
<IPv6-access-list-name>	Specify a Standard or an Extended software IPv6 access-list name. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Default No access list is configured by default.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Examples In the following example, the VLAN interface `vlan2` will only accept MLD joins for groups in the range `ff1e:0db8:0001::/64`:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 access-list standard group1 permit
ff1e:0db8:0001::/64
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld access-group group1
```

In the following example, the VLAN interfaces `vlan2-vlan4` will only accept MLD joins for groups in the range `ff1e:0db8:0001::/64`:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 access-list standard group1 permit
ff1e:0db8:0001::/64
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld access-group group1
```

ipv6 mld immediate-leave

Use this command to minimize the leave latency of MLD memberships.

Use the **no** variant of this command to disable this feature.

Syntax `ipv6 mld immediate-leave group-list <IPv6-access-list-name>`

`no ipv6 mld immediate-leave`

Parameter	Description
<IPv6-access-list-name>	Specify a Standard or an Extended software IPv6 access-list name that defines multicast groups in which the immediate leave feature is enabled. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Default Disabled

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Example The following example shows how to enable the immediate-leave feature on an interface for a specific range of multicast groups. In this example, the router assumes that the group access-list consists of groups that have only one node membership at a time per interface:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld immediate-leave v6grp
awplus(config-if)# exit
```

Related Commands **ipv6 mld last-member-query-interval**

ipv6 mld last-member-query-count

Use this command to set the last-member query-count value.

Use the **no** variant of this command to return to the default on an interface.

Syntax

```
ipv6 mld last-member-query-count <value>
no ipv6 mld last-member-query-count
```

Parameter	Description
<value>	Count value. Valid values are from `2 to 7`.

Default The default last-member query-count value is 2.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld last-member-query-count 3
```

ipv6 mld last-member-query-interval

Use this command to configure the interval at which the router sends MLD group-specific host query messages.

Use the **no** variant of this command to set this frequency to the default.

Syntax

```
ipv6 mld last-member-query-interval <milliseconds>
no ipv6 mld last-member-query-interval
```

Parameter	Description
<milliseconds>	The time delay between successive query messages (in milliseconds). Valid values are from 1000 to 25500 milliseconds.

Default 1000 milliseconds

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Example The following example changes the MLD group-specific host query message interval to 2 seconds:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld last-member-query-interval 2000
```

Related Commands ipv6 mld immediate-leave

ipv6 mld limit

Use this command to configure a limit on the maximum number of group memberships that may be learned. The limit may be set for the switch as a whole, or for a specific interface.

Once the specified group membership limit is reached, all further local-memberships will be ignored.

Optionally, an exception access-list can be configured to specify the group-address(es) that are exempted from being subject to the limit.

Use the **no** variant of this command to unset the limit and any specified exception access-list.

Syntax

```
ipv6 mld limit <limitvalue> [except <IPv6-access-list-name>]
no ipv6 mld limit
```

Parameter	Description
<limitvalue>	<2-512> Maximum number of group membership states.
<IPv6-access-list-name>	Specify a Standard or an Extended software IPv6 access-list name that defines multicast groups, which are exempted from being subject to the configured limit. See **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for supported IPv6 ACLs.

Default The default limit, which is reset by the **no** variant of this command, is the same as maximum number of group membership entries that can be learned with the **ipv6 mld limit** command.

The default limit of group membership entries that can be learned is 512 entries.

Mode Global Configuration and Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols and learned by MLD Snooping.

Examples The following example configures an MLD limit of 100 group-memberships across all VLAN interfaces on which MLD is enabled, and excludes groups in the range `ff1e:0db8:0001::/64` from this limitation:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# ipv6 access-list standard v6grp permit
ff1e:0db8:0001::/64
awplus(config)# ipv6 mld limit 100 except v6grp
```

The following example configures an MLD limit of 100 group-membership states on the VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld limit 100
```

The following example configures an MLD limit of 100 group-membership states on the VLAN interfaces `vlan2-vlan4`:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld limit 100
```

Related Commands **show ipv6 mld groups**

ipv6 mld querier-timeout

Use this command to configure the timeout period before the router takes over as the querier for the interface after the previous querier has stopped querying.

Use the **no** variant of this command to restore the default.

Syntax

```
ipv6 mld querier-timeout <seconds>
no ipv6 mld querier-timeout
```

Parameter	Description
<seconds>	Number of seconds that the router waits after the previous querier has stopped querying before it takes over as the querier. Valid values are from `2 to 65535 seconds`.

Default 255 seconds

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols.

Example The following example configures the router to wait 120 seconds from the time it received the last query before it takes over as the querier for the interface:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld querier-timeout 120
```

Related Commands ipv6 mld query-interval

ipv6 mld query-interval

Use this command to configure the frequency of sending MLD host query messages.

Use the **no** variant of this command to return to the default frequency.

Syntax

```
ipv6 mld query-interval <seconds>
no ipv6 mld query-interval
```

Parameter	Description
<seconds>	Variable that specifies the time delay between successive MLD host query messages (in seconds). Valid values are from `1 to 18000 seconds`.

Default The default query interval is 125 seconds.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols.

Example The following example changes the frequency of sending MLD host-query messages to 2 minutes:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld query-interval 120
```

Related Commands ipv6 mld querier-timeout

ipv6 mld query-max-response-time

Use this command to configure the maximum response time advertised in MLD queries.

Use the **no** variant of with this command to restore the default.

Syntax

```
ipv6 mld query-max-response-time <seconds>
no ipv6 mld query-max-response-time
```

Parameter	Description
<seconds>	Maximum response time (in seconds) advertised in MLD queries. Valid values are from 1 to 240 seconds.

Default 10 seconds

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols.

Example The following example configures a maximum response time of 8 seconds:

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld query-max-response-time 8
```

ipv6 mld robustness-variable

Use this command to change the robustness variable value on an interface.

Use the **no** variant of this command to return to the default on an interface.

Syntax

```
ipv6 mld robustness-variable <value>
no ipv6 mld robustness-variable
```

Parameter	Description
<*value*>	Valid values are from `1 to 7`.

Default The default robustness variable value is 2.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld robustness-variable 3
```

ipv6 mld snooping

Use this command to enable MLD Snooping. When this command is issued in the Global Configuration mode, MLD Snooping is enabled globally for the switch. When this command is issued in Interface mode for a VLAN then MLD Snooping is enabled for the specified VLAN. Note that MLD Snooping is enabled on the VLAN only if it is enabled globally and on the VLAN.

Use the **no** variant of this command to globally disable MLD Snooping in Global Configuration mode, or for the specified VLAN interface in Interface mode.

Note

There is a 100 MLD interface limit when applying MLD commands to multiple VLANs. Only the first 100 VLANs have the required multicast structures added to the interfaces that allow multicast routing. x510 series switches have a 128 MLD group limit for (*, G) and (S,G) entries. See the limits for MLD interfaces depending on the number of VLANs, ports, static and dynamic groups as shown in the relevant product data sheet for your switch.

Syntax

```
ipv6 mld snooping
no ipv6 mld snooping
```

Default By default, MLD Snooping is enabled both globally and on all VLANs.

Mode Global Configuration and Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage For MLD Snooping to operate on particular VLAN interfaces, it must be enabled both globally by using this command in Global Configuration mode, and on individual VLAN interfaces by using this command in Interface Configuration mode (both are enabled by default).

MLD requires memory for storing data structures, as well as the hardware tables to implement hardware routing. As the number of ports, VLANs, static and dynamic groups increases then more memory is consumed. You can track the memory used for MLD with the command:

```
awplus# show memory pools nsm | grep MLD
```

Static and dynamic groups (LACP), ports and VLANs are not limited for MLD. For VLANs, this allows you to configure MLD across more VLANs with fewer ports per VLAN, or fewer VLANs with more ports per VLAN. For LACPs, you can configure MLD across more LACP groups with fewer ports per LACP, or fewer LACP groups with more ports per LACP.

Examples To configure MLD Snooping on the VLAN interface `vlan2`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping
```

To configure MLD Snooping on the VLAN interfaces `vlan2-vlan4`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 mld snooping
```

To disable MLD Snooping for the VLAN interface `vlan2`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config)# no ipv6 mld snooping
```

To disable MLD Snooping for the VLAN interfaces `vlan2-vlan4`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config)# no ipv6 mld snooping
```

To configure MLD Snooping globally for the switch, enter the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 mld snooping
```

To disable MLD Snooping globally for the switch, enter the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 mld snooping
```

ipv6 mld snooping fast-leave

Use this command to enable MLD Snooping fast-leave processing. Fast-leave processing is analogous to immediate-leave processing; the MLD group-membership is removed as soon as an MLD leave group message is received, without sending out a group-specific query.

Use the **no** variant of this command to disable fast-leave processing.

Syntax

```
ipv6 mld snooping fast-leave
no ipv6 mld snooping fast-leave
```

Default MLD Snooping fast-leave processing is disabled.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This MLD Snooping command can only be configured on VLAN interfaces.

Examples This example shows how to enable fast-leave processing on the VLAN interface `vlan2`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping fast-leave
```

This example shows how to enable fast-leave processing on the VLAN interface `vlan2-vlan4`.

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 mld snooping fast-leave
```

ipv6 mld snooping mrouter

Use this command to statically configure the specified port as a Multicast Router interface for MLD Snooping within the specified VLAN.

See detailed usage notes below to configure static multicast router ports when using static IPv6 multicast routes with EPSR, and the destination VLAN is an EPSR data VLAN.

Use the **no** variant of this command to remove the static configuration of the interface as a Multicast Router interface.

Syntax

```
ipv6 mld snooping mrouter interface <port>
no ipv6 mld snooping mrouter interface <port>
```

Parameter	Description
<port>	Specify the name of the port.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This MLD Snooping command statically configures a switch port as a Multicast Router interface.

Note that if static IPv6 multicast routing is being used with EPSR and the destination VLAN is an EPSR data VLAN, then multicast router (mrouter) ports must be statically configured. This minimizes disruption for multicast traffic in the event of ring failure or restoration.

When configuring the EPSR data VLAN, statically configure mrouter ports so that the multicast router can be reached in either direction around the EPSR ring.

For example, if `port1.0.1` and `port1.0.6` are ports on an EPSR data VLAN `vlan101`, which is the destination for a static IPv6 multicast route, then configure both ports as multicast router (mrouter) ports as shown in the example commands listed below:

Output **Figure 56-2: Example ipv6 mld snooping mrouter commands when static IPv6 multicast routing is being used and the destination VLAN is an ESPR data VLAN:**

```
awplus>enable
awplus#configure terminal
awplus(config)#interface vlan101
awplus(config-if)#ipv6 mld snooping mrouter interface port1.0.1
awplus(config-if)#ipv6 mld snooping mrouter interface port1.0.6
```

Examples This example shows how to specify the next-hop interface to the multicast router for VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping mrouter interface
port1.0.5
```

This example shows how to specify the next-hop interface to the multicast router for VLAN interfaces `vlan2-vlan4`:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 mld snooping mrouter interface
port1.0.5
```

Related Commands **ipv6 multicast route**

ipv6 mld snooping querier

Use this command to enable MLD querier operation on a subnet (VLAN) when no multicast routing protocol is configured in the subnet (VLAN). When enabled, the MLD Snooping querier sends out periodic MLD queries for all interfaces on that VLAN.

Use the **no** variant of this command to disable MLD querier configuration.

Syntax

```
ipv6 mld snooping querier
no ipv6 mld snooping querier
```

Mode Interface Configuration for a specified VLAN interface.

Usage This command can only be configured on a single VLAN interface - not on multiple VLANs.

The MLD Snooping querier uses the 0.0.0.0 Source IP address because it only masquerades as an MLD querier for faster network convergence.

The MLD Snooping querier does not start, or automatically cease, the MLD Querier operation if it detects query message(s) from a multicast router. It restarts as an MLD Snooping querier if no queries are seen within the other querier interval.

Do not enable MLD Snooping querier if you have already enabled MLD on your device.

Do not enable MLD Snooping querier on your device and then enable MLD afterwards.

Example

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping querier
```

ipv6 mld snooping report-suppression

Use this command to enable report suppression from hosts for Multicast Listener Discovery version 1 (MLDv1) on a VLAN in Interface Configuration mode.

Use the **no** variant of this command to disable report suppression on a VLAN in Interface Configuration mode.

Syntax

```
ipv6 mld snooping report-suppression
no ipv6 mld snooping report-suppression
```

Default Report suppression does not apply to MLDv2, and is turned on by default for MLDv1 reports.

Mode Interface Configuration for a specified VLAN interface or a range of VLAN interfaces.

Usage This MLD Snooping command can only be configured on VLAN interfaces.

MLDv1 Snooping maybe configured to suppress reports from hosts. When a querier sends a query, only the first report for particular set of group(s) from a host will be forwarded to the querier by the MLD Snooping switch. Similar reports (to the same set of groups) from other hosts, which would not change group memberships in the querier, will be suppressed by the MLD Snooping switch to prevent 'flooding' of query responses.

Examples This example shows how to enable report suppression for MLD reports on VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld snooping report-suppression
```

This example shows how to disable report suppression for MLD reports on VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 mld snooping report-suppression
```

This example shows how to enable report suppression for MLD reports on VLAN interfaces `vlan2-vlan4`:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ipv6 mld snooping report-suppression
```

This example shows how to disable report suppression for MLD reports on VLAN interfaces `vlan2-vlan4`:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# no ipv6 mld snooping report-suppression
```

ipv6 mld static-group

Use this command to statically configure IPv6 group membership entries on an interface. To statically add only a group membership, do not specify any parameters.

Use the **no** variant of this command to delete static group membership entries.

Syntax

```
ipv6 mld static-group <ipv6-group-address>
   [source <ipv6-source-address>|ssm-map] [interface <port>]

no ipv6 mld static-group <ipv6-group-address>
   [source <ipv6-source-address>|ssm-map] [interface <port>]
```

Parameter	Description
<ipv6-group-address>	Specify a standard IPv6 Multicast group address to be configured as a static group member. The IPv6 address uses the format X:X::X:X.
<ipv6-source-address>	Optional. Specify a standard IPv6 source address to be configured as a static source from where multicast packets originate. The IPv6 address uses the format X:X::X:X.
`ssm-map`	Mode of defining SSM mapping. SSM mapping statically assigns sources to MLDv1 groups to translate these (*,G) groups' memberships to (S,G) memberships for use with PIM-SSM.
<port>	Optional. Physical interface. This parameter specifies a physical port. If this parameter is used, the static configuration is applied to just to that physical interface. If this parameter is not used, the static configuration is applied on all ports in the VLAN.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to MLD Snooping on a VLAN interface to statically add groups and/or source records.

Examples The following examples show how to statically add group and/or source records for MLD:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10

awplus # configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
fe80::2fd:6cff:fe1c:b
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
ssm-map
```

The following examples show how to statically add group and/or source records for MLD Snooping on VLAN interface `vlan2`:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
fe80::2fd:6cff:fe1c:b
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
ssm-map
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 interface
port1.0.4
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
fe80::2fd:6cff:fe1c:b interface port1.0.4
```

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 mld static-group ff1e::10 source
ssm-map interface port1.0.4
```

ipv6 mld version

Use this command to set the current MLD protocol version on an interface.

Use the **no** variant of this command to return to the default version on an interface.

Syntax

```
ipv6 mld version <version>
no ipv6 mld version
```

Parameter	Description
<version>	MLD protocol version number. Valid version numbers are 1 and 2

Default The default MLD protocol version number is 2.

Mode Interface Configuration for a VLAN interface.

Usage This command applies to interfaces configured for MLD Layer-3 multicast protocols, MLD Snooping. Note this command is intended for use where there is another querier (when there is another device with MLD enabled) on the same link that can only operate with MLD version 1. Otherwise, the default MLD version 2 is recommended for performance.

Example

```
awplus# configure terminal
awplus(config)# ipv6 forwarding
awplus(config)# ipv6 multicast-routing
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 mld version 1
```

show debugging mld

Use this command to display the MLD debugging modes enabled with the **debug mld** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show debugging mld
```

Mode

Privileged Exec

Example

```
awplus# show debugging mld
```

Output

```
show debugging mld
MLD Debugging status:
  MLD Decoder debugging is on
  MLD Encoder debugging is on
  MLD Events debugging is on
  MLD FSM debugging is on
  MLD Tree-Info-Base (TIB) debugging is on
```

Related Commands

debug mld

show ipv6 mld groups

Use this command to display the multicast groups with receivers directly connected to the router, and learned through MLD.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ipv6 mld groups [<ipv6-address> |<interface>] [detail]
```

Parameter	Description
<ipv6-address>	Optional. Specify Address of the multicast group in format X:X::X:X.
<interface>	Optional. Specify the Interface name for which to display local information.

Mode User Exec and Privileged Exec

Examples The following command displays local-membership information for all interfaces:

```
awplus# show ipv6 mld groups
```

Output

```
MLD Connected Group Membership
Group Address Interface  Uptime  Expires Last Reporter
ff08::1     port1.0.1 00:00:24 stopped fe80::eecd:6dff:fe6b:4783
```

The following command displays local-membership information for all interfaces:

```
awplus# show ipv6 mld groups detail
```

Output

```
MLD Connected Group Membership Details for port1.0.1
Interface:      port1.0.1
Group:          ff08::1
Uptime:         00:00:13
Group mode:     Include ()
Last reporter:  fe80::eecd:6dff:fe6b:4783
Group source list: (R - Remote, M - SSM Mapping, S - Static )
  Source Address                Uptime    v2 Exp    Fwd  Flags
  2001:db8::1                   00:00:13  00:04:07  Yes  R
  2002:db8::3                   00:00:13  00:04:07  Yes  R
```

show ipv6 mld interface

Use this command to display the state of MLD and MLD Snooping for a specified interface, or all interfaces.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 mld interface [<interface>]`

Parameter	Description
<interface>	Interface name.

Mode User Exec and Privileged Exec

Example The following command displays MLD interface status on all interfaces enabled for MLD:

```
awplus# show ipv6 mld interface
```

Output

```
awplus#show ipv6 mld interface
Interface vlan1 (Index 301)
 MLD Enabled, Active, Querier, Version 2 (default)
 Internet address is fe80::215:77ff:fec9:7468
 MLD interface has 0 group-record states
 MLD activity: 0 joins, 0 leaves
 MLD robustness variable is 2
 MLD last member query count is 2
 MLD query interval is 125 seconds
 MLD querier timeout is 255 seconds
 MLD max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
 MLD Snooping is globally enabled
 MLD Snooping is enabled on this interface
 MLD Snooping fast-leave is not enabled
 MLD Snooping querier is enabled
 MLD Snooping report suppression is enabled
```

show ipv6 mld snooping mrouter

Use this command to display the multicast router interfaces, both configured and learned, in a VLAN. If you do not specify a VLAN interface then all the VLAN interfaces are displayed.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 mld snooping mrouter [<interface>]`

Parameter	Description
<interface>	Optional. Specify the name of the VLAN interface. Note: If you do not specify a single VLAN interface, then all VLAN interfaces are shown.

Mode User Exec and Privileged Exec

Examples The following command displays the multicast router interfaces in `vlan2`:

```
awplus# show ipv6 mld snooping mrouter vlan2
```

Output

```
awplus#show ipv6 mld snooping mrouter vlan2
VLAN    Interface    Static/Dynamic
2       port1.0.2    Dynamically Learned
2       port1.0.3    Dynamically Learned
```

The following command displays the multicast router interfaces for all VLAN interfaces:

```
awplus# show ipv6 mld snooping mrouter
```

Output

```
awplus#show ipv6 mld snooping mrouter
VLAN    Interface    Static/Dynamic
2       port1.0.2    Dynamically Learned
2       port1.0.3    Dynamically Learned
3       port1.0.4    Statically Assigned
3       port1.0.5    Statically Assigned
```

show ipv6 mld snooping statistics

Use this command to display MLD Snooping statistics data.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 mld snooping statistics interface <interface>`

Parameter	Description
<interface>	The name of the VLAN interface.

Mode User Exec and Privileged Exec

Example The following command displays MLDv2 statistical information for `vlan1`:

```
awplus# show ipv6 mld snooping statistics interface vlan1
```

Output

```
awplus#show ipv6 mld snooping statistics interface vlan1
MLD Snooping statistics for vlan1
Interface:      port1.0.1
Group:          ff08::1
Uptime:         00:02:18
Group mode:     Include ()
Last reporter:  fe80::eecd:6dff:fe6b:4783
Group source list: (R - Remote, M - SSM Mapping, S - Static )
  Source Address       Uptime    v2 Exp    Fwd  Flags
  2001:db8::1          00:02:18  00:02:02  Yes  R
  2001:db8::3          00:02:18  00:02:02  Yes  R
```

Part 5: Access and Security

- Chapter 57 Access Control Lists Introduction
- Chapter 58 IPv4 Hardware Access Control List (ACL) Commands
- Chapter 59 IPv4 Software Access Control List (ACL) Commands
- Chapter 60 IPv6 Hardware Access Control List (ACL) Commands
- Chapter 61 IPv6 Software Access Control List (ACL) Commands
- Chapter 62 Quality of Service (QoS) Introduction
- Chapter 63 QoS Commands
- Chapter 64 Authentication Introduction and Configuration
- Chapter 65 Authentication Commands
- Chapter 66 AAA Introduction and Configuration
- Chapter 67 AAA Commands
- Chapter 68 802.1X Introduction and Configuration
- Chapter 69 802.1X Commands
- Chapter 70 RADIUS Introduction and Configuration
- Chapter 71 RADIUS Commands
- Chapter 72 Local RADIUS Server Introduction and Configuration
- Chapter 73 Local RADIUS Server Commands
- Chapter 74 TACACS+ Introduction and Configuration

- Chapter 75 TACACS+ Commands
- Chapter 76 Secure Shell (SSH) Introduction
- Chapter 77 Secure Shell (SSH) Configuration
- Chapter 78 Secure Shell (SSH) Commands
- Chapter 79 DHCP Snooping Introduction and Configuration
- Chapter 80 DHCP Snooping Commands

Chapter 57: Access Control Lists Introduction

Introduction 57.2
Overview 57.2
ACL Rules 57.3
ACL Source and Destination Addresses 57.3
ACL Reverse Masking 57.3
Hardware and Software ACL Types 57.4
Defining Hardware MAC ACLs 57.5
Defining Hardware IP ACLs 57.6
Actions for Hardware ACLs 57.7
Attaching Hardware ACLs to Interfaces 57.7
Hardware ACLs and QoS Classifications 57.8
Classifying Your Traffic 57.8
Security ACLs 57.8
QoS ACLs 57.9
Attaching Hardware ACLs Using QoS 57.9
Filtering Hardware ACLs with QoS 57.11
Using QoS Match Commands with TCP Flags 57.11
ACL Filter Sequence Numbers 57.14
ACL Filter Sequence Number Behavior 57.14
ACL Filter Sequence Number Applicability 57.14
ACL Filter Sequence Number Types 57.15
ACL Filter Sequence Configuration 57.18
Creating ACLs in Global Configuration Mode 57.20
Display the ACL configuration details 57.22

Introduction

This chapter describes Access Control Lists (ACLs), and general ACL configuration information.

See **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands** and **Chapter 60, IPv6 Hardware Access Control List (ACL) Commands** for detailed command information and command examples about IPv4 and IPv6 hardware ACLs that are applied directly to interfaces.

See **Chapter 59, IPv4 Software Access Control List (ACL) Commands** and **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for detailed command information and command examples about IPv4 and IPv6 software ACLs as applied to Routing and Multicasting.

See all relevant Routing commands and configurations in **"Layer Three Switching and Routing"** and all relevant Multicast commands and configurations in **"Multicast Applications"**.

Overview

An Access Control List (ACL) is one filter, or a sequence of filters, that are applied to an interface to either block or pass (or when using QoS, apply priority to) packets that match the filter definitions. ACLs are used to restrict network access by hosts and devices and to limit network traffic.

An ACL contains an ordered list of filters. Each filter specifies either permit or deny and a set of conditions the packet must satisfy in order to match the filter. The meaning of permit or deny entries depends on the context in which the ACL is used - either on an inbound or an outbound interface.

When a packet is received on an interface, the switch compares fields in the packet against filters in the ACL to check whether the packet has permission to be forwarded, based on the filter properties. The first match determines whether the switch accepts or rejects the packets. If no entries match, the switch rejects the packets. If there are no restrictions, the switch forwards the packets.

Because filters in an ACL are applied sequentially and their action stops at the first match, it is very important that you apply the filters in the correct order. For example you might want to pass all traffic from VLAN 4 except for that arriving from two selected addresses A and B. Setting up a filter that first passes all traffic from VLAN 4 then denies traffic from addresses A and B will not filter out traffic from A and B if they are members VLAN 4. To ensure that the traffic from A and B is always blocked you should first apply the filter to block traffic from A and B, then apply the filter to allow all traffic from VLAN 4.

You can assign sequence numbers to filters. See **"ACL Filter Sequence Numbers" on page 57.14** for more information.

ACL Rules

- The source or destination address or the protocol of each packet being filtered are tested against the filters in the ACL, one condition at a time (for a permit or a deny filter).
- If a packet does not match a filter then the packet is checked against the next filter in the ACL.
- If a packet and a filter match, the subsequent filters in the ACL are not checked and the packet is permitted or denied as specified in the matched filter.
- The first filter that the packet matches determines whether the packet is permitted or denied. After the first match, no subsequent filters are considered.
- If the ACL denies the address or protocol then the software discards the packet.
- For software ACLs, if no filters match then the packet is dropped.
- For hardware ACLs, if no filters match then the packet is forwarded.
- Checking stops after the first match, so the order of the filters in the ACL is critical. The same permit or deny filter specified in a different order could result in a packet being passed in one situation and denied in another situation.
- One ACL per interface, per protocol, per direction is allowed. However, each ACL assigned per interface, per protocol, per direction may also have multiple filters.
- For inbound ACLs, a permit filter continues to process the packet after receiving it on an inbound interface, and a deny filter discards the packet.

ACL Source and Destination Addresses

Configure source addresses in ACL filters to filter packets coming **from** specified networking devices or hosts. Configure destination addresses in ACL filters to filter packets going **to** specified networking devices or hosts.

ACL Reverse Masking

ACLs uses reverse masking, also referred to as wildcard masking, to indicate to the switch whether to check or ignore corresponding IP address bits when comparing the address bits in an ACL filter to a packet being submitted to the ACL.

Reverse masking for IP address bits specify how the switch treats the corresponding IP address bits. A reverse mask is also called an inverted mask because a 1 and 0 mean the opposite of what they mean in a subnet or a network mask.

- A reverse mask bit 0 means check the corresponding bit value.
- A reverse mask bit 1 means ignore the corresponding bit value.

Hardware and Software ACL Types

Access Control Lists (ACLs) used in AlliedWare Plus™ are separated into two different types, **software ACLs** and **hardware ACLs**. You can define both types as either named or numbered.

Note

The filtering principles applied to software ACLs (those in the range 1 to 2699) are different to those applied to hardware ACLs (those in the range 3000 to 4699).
Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.
Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Numbered ACLs (for hardware and software ACLs)

Numbered ACLs are assigned an ACL number within the range 1 to 4699. ACL numbers are grouped into ranges, where each range denotes a specific functionality. The following table shows the number ranges and functionality that your switch supports.

Table 57-1: ACL numeric ranges and functionality

ACL Number Range	Function
1 to 99	IP standard ACL [1]
100 to 199	IP extended ACL [1]
1300 to 1999	IP standard expanded ACL [1]
2000 to 2699	IP extended expanded ACL [1]
3000 to 3699	Hardware IP ACL
4000 to 4699	Hardware MAC ACL

1. **Software ACLs that use either the ranges 1-99, 100-199, 1300-1999, 2000-2699, or are named ACLs (that use the standard or extended keyword followed by a text string), are used in features such as SNMP, IGMP and OSPF.**

Hardware ACLs

These ACL types are applied directly to an interface, or are used for QoS classifications. They use the following ranges:

- 3000-3699 for Hardware IP ACLs
- 4000-4699 for Hardware MAC ACLs
- named hardware IPv4 ACLs
- named hardware IPv6 ACLs

See **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands** and **Chapter 60, IPv6 Hardware Access Control List (ACL) Commands** for detailed command information and command examples about IPv4 and IPv6 hardware ACLs that are applied directly to interfaces.

Software ACLs

These ACLs types can be either named ACLs, using the standard or extended keyword followed by a text string, or they can use the following ranges:

- 1-99 (IP standard ACL range)
- 100-199 (IP extended ACL range)
- 1300-1999 (IP standard expanded ACL range)
- 2000-2699 (IP extended expanded ACL range)
- named standard IPv4 ACLs
- named extended IPv4 ACLs
- named standard IPv6 ACLs
- named extended IPv6 ACLs

Software ACLs are used in features such as SNMP, PIM, IGMP and OSPF.

See **Chapter 59, IPv4 Software Access Control List (ACL) Commands** and **Chapter 61, IPv6 Software Access Control List (ACL) Commands** for detailed command information and command examples about IPv4 and IPv6 software ACLs as applied to routing and multicasting. See all relevant Routing commands and configurations in **"Layer Three Switching and Routing"** and all relevant multicast commands and configurations in **"Multicast Applications"**.

Defining Hardware MAC ACLs

These are used to filter traffic based on specific source or destination MAC addresses contained within the data frames. They can be applied to ports in the form of access groups.

A MAC access list requires the following components:

- an ACL number in the range 4000-4699
- an action, permit, or deny. See **"Actions for Hardware ACLs" on page 57.7**
- a source MAC address. You can use the format, HHHH.HHHH.HHHH to filter on a specific MAC address (where H is a hexadecimal number), or you can filter on any source MAC address by entering the word "any".
- a source MAC mask. This mask determines which portion of the source MAC address header will be compared with that found in the incoming packets. The mask is configured in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number. In practice each hex number will normally be either 0 (to represent a match) or F (to represent a don't care condition). A mask is not required if the source address is specified as "any".
- a destination MAC address. You can use the format, HHHH.HHHH.HHHH to filter on a specific MAC address (where H is a hexadecimal number), or you can filter on any destination MAC address by entering the word "any".
- a destination MAC mask. This mask determines which portion of the destination MAC address header will be compared with that found in the incoming packets. The mask is configured in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number. In practice each hex number will normally be either 0 (to represent a match) or F (to represent a don't care condition). A mask is not required if the source address is specified as "any".

Example To permit packets coming from a specific MAC address of 0030.841A.1234 and with any destination address:

```
awplus# configure terminal
awplus(config)# access-list 4000 permit 0030.841A.1234
0000.0000.0000 any
```

Defining Hardware IP ACLs

Hardware IP ACLs are used to filter traffic based on specific source or destination IP addresses contained within the data frames. They can be applied to ports in the form of access groups.

An IP access list requires the following components:

- an ACL number in the range 3000-3699
- an action, see **"Actions for Hardware ACLs" on page 57.7**
- a packet type:
  - « IP: This matches any type of IP packet. A source and destination address must be specified, although they can be "any".
  - « ICMP: This matches ICMP packets. A source and destination address must be specified, although they can be "any". An ICMP type can optionally be specified after the destination address.
  - « TCP: This matches TCP packets. A source and destination address must be specified, although they can be "any". After the source address, a source port can optionally be specified and after the destination address a destination port can optionally be specified. The port matching can be done using **eq** (equal to), **gt** (greater than), **lt** (less than), **ne** (not equal to), or **range** (for a range of ports, which requires a start port and an end port).
  - « UDP: This matches UDP packets and has the same options as TCP.
  - « proto: This allows any IP protocol type to be specified (e.g. 89 for OSPF). A source and destination address must be also specified, although they can be "any".

For example, to match (and permit) any type of IP packet containing a destination address of 192.168.1.1

```
awplus(config)# access-list 3000 permit ip any 192.168.1.1/32
```

To match (and permit) an ICMP packet with a source address of 192.168.x.x and an ICMP code of 4

```
awplus(config)# access-list 3001 permit icmp 192.168.0.0/16
any icmp-type 4
```

To match a TCP packet with a source address of 192.168.x.x, source port of 80 and a destination port from 100 to 150:

```
awplus(config)# access-list 3002 permit tcp 192.168.0.0/16 eq
80 any range 100 150
```

To match a UDP packet with a source address of 192.168.x.x, a destination address of 192.168.1.x, and a destination port greater than 80:

```
awplus(config)# access-list 3003 permit udp 192.168.0.0/16
                192.168.1.0/24 gt 80
```

To match to any OSPF packet:

```
awplus(config)# access-list 3004 permit proto 89 any any
```

Note that an IP address mask can be specified using either of the following notations:

- "A.B.C.D/M": This is the most common; e.g. 192.168.1.0/24
- "A.B.C.D A.B.C.D": 192.168.1.1 0.0.0.0 is the same as 192.168.1.1/32 and 192.168.1.1 255.255.255.255 is the same as "any"
- "host A.B.C.D": This is the same as A.B.C.D/32

Actions for Hardware ACLs

The following actions are available for Hardware ACLs:

Parameter	Action
deny	Discard the packet
permit	Allow the packet
copy-to-cpu	Send a copy of the packet to the CPU and forward it as well. This is the same as copy, forward in AW hardware filters.
send-to-cpu	Send the packet to the CPU and do not forward it. This is the same as copy, discard in AlliedWare hardware filters.
send-to-mirror	Send the packet to the mirror port so packets are not switched.
copy-to-mirror	Send a copy of the packet to the mirror port and forward it as well.

Attaching Hardware ACLs to Interfaces

A hardware ACL is attached directly to a switchport using the **access-group** command. For example, to permit traffic from 192.168.1.x, but discard from 192.168.x.x:

```
awplus# configure terminal
awplus(config)# access-list 3000 permit ip 192.168.1.0/24
                any
awplus(config)# access-list 3001 deny ip 192.168.0.0/24 any
awplus(config)# interface port1.0.1
awplus(config-if)# access-group 3000
awplus(config-if)# access-group 3001
```

Hardware ACLs and QoS Classifications

Interface ACLs and QoS policies can both be attached to the same port. Where this is done, packets received on the port will be matched against the ACLs first.

The interface ACLs and QoS classifications are implemented by taking the first matching filter and applying the action defined for that filter. All subsequent matches in the table are then ignored. Thus, because ACLs are also matched first, if the matching ACL has a permit action, the packet is forwarded due to that rule's action and any subsequent QoS rules are bypassed.

You can also apply permit rules using QoS.

For example, you might want to permit a source IP address of 192.168.1.x, but block everything else on 192.168.x.x.

In this case you could create both the permit and deny rules using QoS.

Classifying Your Traffic

Classification is the process of **filtering** and **marking**. Filtering involves sorting your data into appropriate traffic types. Marking involves tagging the data so that downstream ports and routers can apply appropriate service policy rules.

There are two reasons to classify data:

1. To provide network security (Security ACLs)
2. To apply service quality criteria QoS.

Security ACLs

The main application of security ACLs is to block undesired traffic. Other applications include:

- copy-to-cpu
- copy-to-mirror
- send-to-cpu
- send-to-mirror

For more information on these applications see **"Actions for Hardware ACLs" on page 57.7**

QoS ACLs

When using ACLs though QoS, the same classification and action abilities are available, but QoS has some additional fields that it can match on (see Match Commands) and also provides the ability to perform metering, marking and remarking on packets that match the filter definitions.

The action used by a QoS class-map is determined by the ACL that is attached to it. If no ACL is attached, it uses the permit action. If an ACL is not required by the class-map (for example, only matching on the VLAN) and a deny action is required, a MAC ACL should be added with `any` for source address and `any` for destination address.

The following example creates a class-map with will deny all traffic on VLAN 2:

```
awplus(config)# access-list 4000 deny any any
awplus(config)# class-map cmap1
awplus(config-cmap)# match access-group 4000
awplus(config-cmap)# match vlan 2
```

The default class-map matches to all traffic and so cannot have any match or ACL commands applied to it. The action for this class-map is set via the default-action command and is `permit` by default. It can be changed to `deny` by using the following commands:

```
awplus(config)# policy-map pmap1
awplus(config-pmap)# default-action deny
```

For more information on applying QoS filtering, see **"Classifying Your Data" on page 62.7**.

Attaching Hardware ACLs Using QoS

The same functionality can be achieved using QoS, by attaching the ACL to a class-map, attaching the class-map to a policy-map and attaching the policy-map to a port:

Step 1: Enable QoS on the switch

```
awplus(config)# mls qos enable
```

Step 2: Create access lists

Create ACL `3000` to permit all packets from the 192.168.1 subnet:

```
awplus(config)# access-list 3000 permit ip 192.168.1.0/24 any
```

Create ACL `3001` to deny all packets from the 192.168.0 subnet.:

```
awplus(config)# access-list 3001 deny ip 192.168.0.0/24 any
```

Step 3: Attach access-groups to class-maps

Attach ACL `3000` to the class-map `cmap1`:

```
awplus(config)# class-map cmap1
awplus(config-cmap)# match access-group 3000
awplus(config-cmap)# exit
```

Attach ACL `3001` to the same class-map (`cmap2`):

```
awplus(config-cmap)# match access-group 3001
awplus(config-cmap)# exit
```

Step 4: Attach class-maps to policy-maps

Attach the class-map `cmap1` to policy-map `pmap1`:

```
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# exit
```

Add the class-map `cmap2` to the policy-map `pmap1`:

```
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# exit
```

Return to Global Configuration mode:

```
awplus(config-pmap)# exit
```

Step 5: Add policy-maps to ports

Add policy-map `pmap1` to `port1.0.1`:

```
awplus(config)# interface port1.0.1
awplus(config-if)# service-policy input pmap1
```

Note that multiple interface ACLs can be attached to the same port, or either type and can be interleaved. The order of matching is based on the order in which the ACLs were attached to the port. Only one ACL can be attached to a class-map, but multiple class-maps can be attached to a policy-map. Interface ACLs can be attached to the same port as a QoS policy, with the interface ACLs being matched first as described at the beginning of the Classification section.

Filtering Hardware ACLs with QoS

Another reason for using QoS rather than interface ACLs is that QoS provides a lot more fields on which to match. These are accessed through the match commands in config-cmap mode.

Config-cmap mode describes the fields that can be matched on. Only one of each type can be matched, with the exception of tcp-flags (see below for classification). If multiple matches are specified, they are ANDed together.

The following example shows how you can match a packet on VLAN 2, that has a source IP address of 192.168.x.x and a DSCP of 12:

Create ACL `3000` to permit all packets from the 192.168 subnet.:

```
awplus# configure terminal
awplus(config)# access-list 3000 permit ip 192.168.0.0/16 any
```

Apply ACL 3000 to the class-map cmap1 and add the matching criteria of VLAN 2 and DSCP 12:

```
awplus(config)# class-map cmap1
awplus(config-cmap)# match access-group 3000
awplus(config-cmap)# match vlan 2
awplus(config-cmap)# match dscp 12
awplus(config-cmap)# exit
```

Using QoS Match Commands with TCP Flags

Usually, if multiple matches of the same type are specified, the matching process will apply to the last match that you specified. For TCP flags however, the arguments are ANDed together. For example, the following series of commands will match on a packet that has ack, syn and fin set:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags ack
awplus(config-cmap)# match tcp-flags syn
awplus(config-cmap)# match tcp-flags fin
awplus(config-cmap)# exit
```

The following commands will achieve the same result:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags ack syn fin
awplus(config-cmap)# exit
```

Note that the matching is looking to see whether "any" of the specified flags are set. There is no checking for whether any of these flags are unset. Therefore the following commands will match on a packet in any of the following combinations of syn and ack status flags as shown in the following table:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags syn
awplus(config-cmap)# exit
```

Syn	Ack	Match on Packet
Set	Set	Yes
Set	Unset	Yes
Unset	Set	No
Unset	Unset	No

If you want to drop packets with syn only, but not with ack and syn, the following two class-maps can be used (note that ACL 4000 is used to apply a drop action as described in **"Actions for Hardware ACLs" on page 57.7**):

Step 1: Create access lists

Create ACL `4000` to deny all packets with `any` source or destination address:

```
awplus# configure terminal
awplus(config)# access-list 4000 deny any any
```

Step 2: Create class-maps

Create the class-map `cmap1` and configure it to match on the TCP flags, `ack` and `syn`:

```
awplus(config)# class-map cmap1
awplus(config-cmap)# match tcp-flags ack syn
awplus(config-cmap)# exit
```

Create the class-map `cmap2` and configure it to match on the TCP flag, `syn`:

```
awplus(config)# class-map cmap2
awplus(config-cmap)# match tcp-flags syn
```

Step 3: Apply access-groups to class-maps

Apply ACL `4000` to this class-map (i.e. to `cmap2`):

```
awplus(config-cmap)# match access-group 4000
awplus(config-cmap)# exit
```

Step 4: Create policy-maps

Create the policy-map pmap1 and associate it with cmap1:

```
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# exit
```

Step 5: Associate class-maps with policy-maps

Associate `cmap2` with this policy-map (`pmap1`):

```
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# exit
```

ACL Filter Sequence Numbers

To help you manage ACLs you can apply sequence numbers to filters. This allows you to remove filters from named and numbered ACLs without having to reconfigure an ACL.

The ability to add sequence numbers to filters simplifies updates through the ability to position a filter within an ACL. When you add a new filter, you can specify a sequence number to position the filter in the ACL and you can also remove a current filter in an ACL by specifying a sequence number.

ACL Filter Sequence Number Behavior

- If filters with no sequence numbers are applied then the first filter is assigned a sequence number of 10, and successive filters are incremented by 10. Sequence numbers are generated automatically if they are not specified at entry.
- The maximum filter sequence number is 65535. If the sequence number exceeds this maximum, the command will not be recognized and will show the error message:
  `% Unrecognized command`
- If you enter a filter without a sequence number it is assigned a sequence number that is 10 greater than the last sequence number and is placed at the end of the ACL.
- If you enter a filter that matches an already existing filter then the first filter is overwritten with the subsequent filter.
- ACL sequence numbers determine the order of execution of filters in an ACL. Filters in a ACL with a lower value sequence number are executed before filters with a higher value.
- Output from **show running-config** displays ACL entries without filter sequence numbers. Output from relevant **show** commands displays ACL entries with their sequence numbers.
- ACL sequence numbers are re-numbered upon switch restart following a **reload** command, or after powering off and powering on the switch. ACL sequence numbers are renumbered starting from 10 and increment by 10 for each filter. See the sample output in the configuration section that follows for an illustration of this behavior. No ACL sequence number re-number command is available to perform this action.
- The ACL sequence number feature works with numbered and named standard and extended IPv4 and IPv6 access lists, plus named hardware IPv4 and IPv6 access lists
- The name of an access list can be designated as a number. Number in named ACLs must not exist within the range of designated numbered ACLs. (where <1-99> and <1300-1999> are standard numbered ACLs, <100-199> and <2000-2699> are extended numbered ACLs, <3000-3699> and <4000-4699> are hardware numbered ACLs).

ACL Filter Sequence Number Applicability

The ACL sequence number support feature is available with numbered and named standard and extended IPv4 and IPv6 ACLs, and the named hardware IPv4 and IPv6 ACLs.

Numbered standard ACLs are available in the range `<1-99>` and `<1300-1999>`, which permit or deny source addresses to control packets coming from network devices or hosts, in software.

Numbered extended ACLs are available in the range <100-199> and <2000-2699>, which permit or deny source addresses and destination addresses (plus ICMP, TCP, UDP messages) to control packets coming from and going to network devices or hosts.

Named hardware IPv4 and IPv6 ACLs are available which permit or deny IP and MAC source and destination addresses plus VLAN IDs to control packets coming from and going to network device and hosts. Named hardware IPv4 and IPv6 ACLs use the ACL sequence number support feature for ACL revision.

The ACL sequence number support feature is available for use with named hardware IPv4 and IPv6 ACLs, but this feature is not available for use with the numbered hardware IPv4 ACLs.

Numbered hardware ACLs are available in the range <3000-3699>, which permit or deny IP source addresses, IP destination addresses, and VLAN IDs to control packets coming from and going to network devices and hosts, in hardware.

Numbered hardware ACLs are available in the range <4000-4699>, which permit or deny MAC source addresses, MAC destination addresses, and VLAN IDs to control packets coming from and going to network devices and hosts, in hardware.

ACL Filter Sequence Number Types

There are ACL filter sequence numbers available for the following types of ACLs:

ACL Type	ACL Command Syntax
IPv4 Standard Numbered ACLs	**access-list <1-99>** **access-list <1300-1999>**
IPv4 Extended Numbered ACLs	**access-list <100-199>** **access-list <2000-2699>**
IPv4 Standard Named ACLs	**access-list standard <*name*>**
IPv4 Extended Named ACLs	**access-list extended <*name*>**
IPv4 Hardware Named ACLs	**access-list hardware <*name*>**
IPv6 Standard Named ACLs	**ipv6 access-list standard <*name*>**
IPv6 Extended Named ACLs	**ipv6 access-list extended <*name*>**
IPv6 Hardware Named ACLs	**ipv6 access-list <*name*>**

Note that ACL sequence number support for these ACL commands is optional not required. An ACL sequence number will be added automatically, starting at 10 and incrementing by 10.

ACL commands without ACL filter sequence numbers

ACL filter sequence numbers are not available for numbered hardware ACL commands:

access-list <3000-3699>

access-list <4000-4699>

ACL filter sequence number entry examples

See the below CLI entry examples for prompt sub-modes for ACL filters after ACL commands:

- To create an IPv4 Standard ACL and then define ACL filters at the IPv4 Standard ACL Configuration mode prompt **awplus(config-ip-std-acl)#**, enter the following commands:

```
awplus(config)# access-list 1
awplus(config-ip-std-acl)# permit 192.168.1.0 0.0.0.255

awplus(config)# access-list standard std_name
awplus(config-ip-std-acl)# permit 192.168.1.0/24
```

- To create an IPv4 Extended ACL and then define ACL filters at the IPv4 Extended ACL Configuration mode prompt **awplus(config-ip-ext-acl)#**, enter the following commands:

```
awplus(config)# access-list 100
awplus(config-ip-ext-acl)# permit ip 192.168.1.0 0.0.0.255
                           192.168.2.0 0.0.0.255

awplus(config)# access-list extended ext_name
awplus(config-ip-ext-acl)# permit ip 192.168.1.0 0.0.0.255
                           192.168.2.0 0.0.0.255
```

- To create an IPv4 Hardware ACL and then define ACL filters at the IPv4 Hardware ACL Configuration mode prompt **awplus(config-ip-hw-acl)#**, enter the following commands:

```
awplus(config)# access-list hardware hw_name
awplus(config-ip-hw-acl)# permit ip 192.168.1.0 0.0.0.255
                           192.168.2.0 0.0.0.255
```

- To create an IPv6 Standard ACL and then define ACL filters at the IPv6 Standard ACL Configuration mode prompt **awplus(config-ipv6-std-acl)#**, enter the following commands:

```
awplus(config)# ipv6 access-list standard ipv6_st-
                d_name
awplus(config-ipv6-std-acl)# permit 2001:db8::/64
```

- To create an IPv6 Extended ACL and then define ACL filters at the IPv6 Extended Configuration mode prompt **awplus(config-ipv6-ext-acl)#**, enter the following commands:

```
awplus(config)# ipv6 access-list extended ipv6_ex-
                t_name
awplus(config-ipv6-ext-acl)# permit ip 2001:db8::/64
                2001:db9::/64
```

- To create an IPv6 Hardware ACL and then define ACL filters at the IPv6 Hardware Configuration mode prompt **awplus(config-ipv6-hw-acl)**#, enter the following commands:

```
awplus(config)# ipv6 access-list ipv6_hw_name
awplus(config-ipv6-hw-acl)# permit ip 2001:db8::/64 2001:db9::/
                64
```

ACL Filter Sequence Configuration

First create a named or numbered ACL to enter ACL filters in the ACL sub-modes available:

Step 1: Create a new ACL and add a new filter

Create ACL `10` and then add a new filter to the access-list to permit all packets from the `192.168.1` subnet:

```
                    awplus# configure terminal
             awplus(config)# access-list 10
  awplus(config-ip-std-acl)# permit 192.168.1.0 0.0.0.255
  awplus(config-ip-std-acl)# end
                    awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
```

Step 2: Add another filter to the ACL

Append to, or add at the end of, ACL `10` a new filter to deny all packets from the `192.168.2` subnet:

```
                    awplus# configure terminal
             awplus(config)# access-list 10
  awplus(config-ip-std-acl)# deny 192.168.2.0 0.0.0.255
  awplus(config-ip-std-acl)# end
                    awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   20 deny   192.168.2.0, wildcard bits 0.0.0.255
```

Note that if you add a filter to an ACL without specifying a sequence number the new filter is automatically assigned a sequence number. Sequence numbers are assigned in multiples of ten from the sequence number of the last filter.

Step 3: Insert a filter into the ACL

Insert a new filter with the sequence number 15 into ACL 10 to permit packets from the 192.168.3 subnet:

```
                    awplus# configure terminal
             awplus(config)# access-list 10
awplus(config-ip-std-acl)# 15 permit 192.168.3.0 0.0.0.255
awplus(config-ip-std-acl)# end
                    awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
   20 deny   192.168.2.0, wildcard bits 0.0.0.255
```

The new filter has precedence over the filter with the sequence number 20.

Step 4: Remove a filter from the ACL by specifying a filter pattern

Remove the filter with the IP address 192.168.2 from ACL 10:

```
                    awplus# configure terminal
             awplus(config)# access-list 10
awplus(config-ip-std-acl)# no deny 192.168.2.0 0.0.0.255
awplus(config-ip-std-acl)# end
                    awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
```

Step 5: Remove a filter from the ACL by specifying a sequence number

Remove the filter with the sequence number 10 from ACL 10:

```
                    awplus# configure terminal
             awplus(config)# access-list 10
awplus(config-ip-std-acl)# no 10
awplus(config-ip-std-acl)# end
                    awplus# show access-list
```

```
Standard IP access list 10
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
```

Creating ACLs in Global Configuration Mode

You can add new filters in **Global Configuration** mode with the **access-list (standard numbered)** command on page 59.29. In this mode the filters are assigned a sequence number corresponding to the order in which they are entered, i.e. the first filter entered has higher precedence in the ACL.

Step 1: Add filters with the access-list command

Add filters to ACL `10` using the **access-list** command:

```
awplus# configure terminal
awplus(config)# access-list 10 permit 192.168.1.0 0.0.0.255
awplus(config)# access-list 10 deny 192.168.2.0 0.0.0.255
awplus(config)# end
awplus# show access-list 10
```

```
Standard IP access list 10
   15 permit 192.168.3.0, wildcard bits 0.0.0.255
   20 permit 192.168.1.0, wildcard bits 0.0.0.255
   30 deny 192.168.2.0, wildcard bits 0.0.0.255
```

You can then enter the **IPv4 Standard ACL Configuration** mode and use the **(access-list standard numbered filter)** command on page 59.33 to specify sequence numbers to reorder the filters.

Step 2: Reorder the filters

Reorder the filters in ACL `10` by specifying a sequence number for each filter. The specified sequence number will overwrite the previous sequence number assigned to the filter:

```
awplus# configure terminal
awplus(config)# access-list 10
awplus(config-ip-std-acl)# 1021 permit 192.168.1.0 0.0.0.255
awplus(config-ip-std-acl)# 3333 permit 192.168.3.0 0.0.0.255
awplus(config-ip-std-acl)# 2772 deny 192.168.2.0 0.0.0.255
awplus(config-ip-std-acl)# end
awplus# show access-list 10
```

```
Standard IP access list 10
 1021 permit 192.168.1.0, wildcard bits 0.0.0.255
 2772 deny   192.168.2.0, wildcard bits 0.0.0.255
 3333 permit 192.168.3.0, wildcard bits 0.0.0.255
```

Step 3: Copy the running-config file into the startup-config file

Copy the running-config into the file set as the current startup-config file and then reload the device. Before the reload occurs, you will receive a confirmation request saying: "`reboot system? (y/n):`".

When the device has reboot you can then enter **Global Configuration** mode and use the **show access-list (IPv4 Software ACLs)** command to display ACL `10`:

```
awplus(config)# exit
awplus# copy running-config startup-config
awplus# reload
awplus# show access-list 10
```

```
Standard IP access list 10
   10 permit 192.168.1.0, wildcard bits 0.0.0.255
   20 deny   192.168.2.0, wildcard bits 0.0.0.255
   30 permit 192.168.3.0, wildcard bits 0.0.0.255
```

After the device has reboot the sequence numbers of the filters in the ACL have been reassigned incrementing from 10.

Display the ACL configuration details

Display the running system status and configuration details for ACLs:

```
awplus# show running-config access-list
```

```
!
access-list 1 deny 10.1.1.0 0.0.0.255
access-list 1 permit any
access-list 2
access-list 5
access-list 10 permit 192.168.1.0 0.0.0.255
access-list 10 deny 192.168.2.0 0.0.0.255
access-list 10 permit 192.168.3.0 0.0.0.255
access-list 20
access-list 25 permit 10.1.2.0 0.0.0.255
access-list 25 deny 192.168.1.0 0.0.0.255
access-list 50
access-list 95 permit any
access-list 100
access-list 1300
access-list 2000
access-list extended acl
access-list extended my-list
access-list extended name
access-list extended name1
access-list standard name3
ipv6 access-list extended ipv6_acl
ipv6 access-list standard ipv6_acl2
ipv6 access-list extended my-ipv6-list
ipv6 access-list extended my-list
ipv6 access-list standard my-new-list
ipv6 access-list standard name
ipv6 access-list standard name1 deny any
ipv6 access-list extended name5
ipv6 access-list standard name6
access-list hw_acl
access-list icmp
access-list my-hw-list
access-list name2
access-list name4
!
```

For more information see **show running-config access-list** command on page 7.39.

Chapter 58: IPv4 Hardware Access Control List (ACL) Commands

Introduction 58.2
IPv4 Hardware Access List Commands and Prompts 58.3
Command List 58.4
access-group 58.4
access-list (hardware IP numbered) 58.6
access-list (hardware MAC numbered) 58.15
access-list hardware (named) 58.18
(access-list hardware ICMP filter) 58.20
(access-list hardware IP protocol filter) 58.23
(access-list hardware MAC filter) 58.28
(access-list hardware TCP UDP filter) 58.31
commit (IPv4) 58.34
show access-list (IPv4 Hardware ACLs) 58.35
show interface access-group 58.37

Introduction

This chapter provides an alphabetical reference of IPv4 Hardware Access Control List (ACL) commands. It contains detailed command information and command examples about IPv4 hardware ACLs, which are applied directly to interfaces using the **access-group** command

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself.

Notes

- Text in parenthesis in command names indicates usage not keyword entry. For example, **access-list hardware (named)** indicates named IPv4 hardware ACLs entered as `access-list hardware` <*name*> where <*name*> is a placeholder not a keyword.
- Parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI, such as **(access-list standard numbered filter)** represents command entry in the format shown in the syntax `[<sequence-number>] {deny|permit} {<source>|host <host-address>|any}`.
- Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

References

For descriptions of ACLs, and further information about rules when applying them, see the **"ACL Rules" on page 57.3**.

For more information about ACL Filters, see the following references:

- **"ACL Filter Sequence Numbers" on page 57.14**
- **"ACL Filter Sequence Number Behavior" on page 57.14**
- **"Access Control Lists Introduction" on page 57.1**

For more information on link aggregation see the following references:

- **Chapter 22, Link Aggregation Introduction and Configuration**
- **Chapter 23, Link Aggregation Commands**

IPv4 Hardware Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table **"IPv4 Hardware Access List Commands and Prompts"** shows the CLI prompts at which ACL commands are entered.

Table 58-1: IPv4 Hardware Access List Commands and Prompts

Command Name	Command Mode	Prompt
show interface access-group	Privileged Exec	`awplus#`
show access-list (IPv4 Hardware ACLs)	Privileged Exec	`awplus#`
show interface access-group	Privileged Exec	`awplus#`
access-group	Global Configuration	`awplus(config)#`
access-list (hardware IP numbered)	Global Configuration	`awplus(config)#`
access-list (hardware MAC numbered)	Global Configuration	`awplus(config)#`
access-list hardware (named)	Global Configuration	`awplus(config)#`
access-group	Interface Configuration	`awplus(config-if)#`
(access-list hardware ICMP filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
(access-list hardware IP protocol filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
(access-list hardware MAC filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
(access-list hardware TCP UDP filter)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`
commit (IPv4)	IPv4 Hardware ACL Configuration	`awplus(config-ip-hw-acl)#`

Command List

access-group

This command adds or removes a hardware-based access-list to or from a switch port interface. The number of hardware numbered and named access-lists that can be added to a switch port interface is determined by the available memory in hardware-based packet classification tables.

This command works in Interface Configuration mode to apply hardware access-lists to selected switch port interfaces.

The **no** variant of this command removes the selected access-list from an interface.

Syntax

```
access-group [<3000-3699>|<4000-4699>|<hardware-access-list-name>]
no access-group [<3000-3699>|4000-4699|<hardware-access-list-name>]
```

Parameter	Description
<3000-3699>	Hardware IP access-list.
<4000-4699>	Hardware MAC access-list.
<hardware-access-list-name>	The hardware access-list name.

Mode Interface Configuration for a switch port interface

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create an IP access-list that applies the appropriate permit/deny requirements with the **access-list (hardware IP numbered)** command on page 58.6, the **access-list (hardware MAC numbered)** command on page 58.15 or the **access-list hardware (named)** command on page 58.18. Then use this command to apply this hardware access-list to a specific port or port range. Note that this command will apply the access-list only to incoming data packets.

To apply ACLs to an LACP aggregated link, apply it to all the individual switch ports in the aggregated group. To apply ACLs to a static channel group, apply it to the static channel group itself. An ACL can even be applied to a static aggregated link that spans more than one switch instance (**Chapter 23, Link Aggregation Commands**).

Note that you cannot apply software standard and extended numbered ACLs to switch port interfaces with the access-group command. This command will only apply hardware ACLs.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To add the numbered hardware access-list `3005` to switch port interface `port1.0.1`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# access-group 3005
```

To add the named hardware access-list `hw-acl` to switch port interface `port1.0.2`, enter the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# access-group hw-acl
```

To apply an ACL to static channel group 2 containing switch `port1.0.5` and `port1.0.6`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.5-1.0.6
awplus(config-if)# static-channel-group 2
awplus(config)# interface sa2
awplus(config-if)# access-group 3000
```

Related Commands
access-list hardware (named)
access-list (hardware IP numbered)
access-list (hardware MAC numbered)
show interface access-group

access-list (hardware IP numbered)

This command creates an access-list for use with hardware classification, such as QoS. The access-list will match on either TCP or UDP type packets that have the specified source and destination IP addresses and Layer 4 port values or ranges. The parameter **any** may be specified if an address does not matter and the port values are optional.

The **no** variant of this command removes the previously specified IP hardware access-list.

Syntax [ip]

```
access-list <3000-3699> {deny|permit|copy-to-cpu|copy-to-mirror|
    send-to-mirror|send-to-cpu} ip <source> <destination>
```

Syntax [icmp]

```
access-list <3000-3699> {deny|permit|copy-to-cpu|copy-to-mirror|
    send-to-mirror|send-to-cpu} icmp <source> <destination>
    [icmp-type <type-number>]

no access-list <3000-3699>
```

Table 58-2: Parameters in the access-list (hardware IP numbered) command - ip|icmp

Parameter	Description	
<3000-3699>	Hardware IP access-list number.	
deny	Access-list rejects packets that match the source and destination filtering specified with this command.	
permit	Access-list permits packets that match the source and destination filtering specified with this command.	
copy-to-cpu	Specify packets to copy to the CPU.	
copy-to-mirror	Specify packets to copy to the mirror port.	
send-to-mirror	Specify packets to send to the mirror port.	
send-to-cpu	Specify packets to send to the CPU.	
icmp	ICMP packet.	
ip	IP packet.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	any	Matches any source IP address.
	host *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.

Table 58-2: Parameters in the access-list (hardware IP numbered) command - ip|icmp

Parameter (cont.)	Description (cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
`icmp-type`	Matches only a specified type of ICMP messages. This is valid only when the filtering is set to match ICMP packets.	
<type-number>	The ICMP type, as defined in RFC792 and RFC950. Specify one of the following integers to create a filter for the ICMP message type:	
	0	Echo replies.
	3	Destination unreachable messages.
	4	Source quench messages.
	5	Redirect (change route) messages.
	8	Echo requests.
	11	Time exceeded messages.
	12	Parameter problem messages.
	13	Timestamp requests.
	14	Timestamp replies.
	15	Information requests.
	16	Information replies.
	17	Address mask requests.
	18	Address mask replies.

Syntax [tcp|udp]

```
access-list <3000-3699> {copy-to-cpu|copy-to-mirror|send-to-mirror|
    deny|permit|send-to-cpu} {tcp|udp} <source> {eq <sourceport>|
    lt <sourceport>|gt <sourceport>|ne <sourceport>|
    [range <start-range> <end-range>} <destination> [eq <destport>|
    lt <destport>|gt <destport>|ne <destport>]
    [range <start-range> <end-range>]

no access-list <3000-3699>
```

Table 58-3: Parameters in the access-list (hardware IP numbered) command - tcp|udp

Parameter	Description	
<3000-3699>	Hardware IP access-list.	
copy-to-cpu	Specify packets to copy to the CPU.	
copy-to-mirror	Specify packets to copy to the mirror port.	
send-to-mirror	Specify packets to send to the mirror port.	
deny	The access-list rejects packets that match the type, source, and destination filtering specified with this command.	
permit	The access-list permits packets that match the type, source, and destination filtering specified with this command.	
send-to-cpu	Specify packets to send to the CPU.	
tcp	The access-list matches only TCP packets.	
udp	The access-list matches only UDP packets.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	any	Matches any source IP address.
	host *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering 192.168.1.1 0.0.0.255 is the same as entering 192.168.1.1/24.

Table 58-3: Parameters in the access-list (hardware IP numbered) command - tcp|udp

Parameter (cont.)	Description (cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/`*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
<sourceport>	The source (TCP or UDP) port number, specified as an integer between 0 and 65535.	
`range`	Range of port numbers.	
<start-range>	Port number at start of range *<0-65535>*.	
<end-range>	Port number at end of range *<0-65535>*.	
<destport>	The destination (TCP or UDP) port number, specified as an integer between 0 and 65535.	
`eq`	Matches port numbers that are equal to the port number specified immediately after this parameter.	
`lt`	Matches port numbers that are less than the port number specified immediately after this parameter.	
`gt`	Matches port numbers that are greater than the port number specified immediately after this parameter.	
`ne`	Matches port numbers that are not equal to the port number specified immediately after this parameter.	

Syntax [proto]

```
access-list <3000-3699> {copy-to-cpu|copy-to-mirror|send-to-mirror|
    deny|permit|send-to-cpu}
    proto <ip-protocol> <source> <destination>

no access-list <3000-3699>
```

Table 58-4: Parameters in the access-list (hardware IP numbered) command - proto

Parameter	Description	
<3000-3699>	Hardware IP access-list.	
`copy-to-cpu`	Specify packets to copy to the CPU.	
`copy-to-mirror`	Specify packets to copy to the mirror port.	
`send-to-mirror`	Specify packets to send to the mirror port	
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.	
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.	
`send-to-cpu`	Specify packets to send to the CPU.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> <reverse-mask>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

Table 58-4: Parameters in the access-list (hardware IP numbered) command - proto

Parameter (cont.)	Description (cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/`*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
`proto`	Matches only a specified type of IP Protocol <1-255>.	
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**)	
	Protocol Number	**Protocol Description [RFC Reference]**
	1	Internet Control Message [RFC792]
	2	Internet Group Management [RFC1112]
	3	Gateway-to-Gateway [RFC823]
	4	IP in IP [RFC2003]
	5	Stream [RFC1190] [RFC1819]
	6	TCP (Transmission Control Protocol) [RFC793]
	8	EGP (Exterior Gateway Protocol) [RFC888]
	9	IGP (Interior Gateway Protocol) [IANA]
	11	Network Voice Protocol [RFC741]
	17	UDP (User Datagram Protocol) [RFC768]
	20	Host monitoring [RFC869]
	27	RDP (Reliable Data Protocol) [RFC908]
	28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
	29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]

Table 58-4: Parameters in the access-list (hardware IP numbered) command - proto

Parameter (cont.)	Description (cont.)	
<ip-protocol> (cont.)	30	Bulk Data Transfer Protocol [RFC969]
	33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
	48	DSR (Dynamic Source Routing Protocol) [RFC4728]
	50	ESP (Encap Security Payload) [RFC2406]
	51	AH (Authentication Header) [RFC2402]
	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	OSPFIGP [RFC1583]
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage This command creates an access-list for use with hardware classification, such as when applying QoS. This command can be used to match ICMP packets, IP protocols, or TCP/UDP packets.

For ICMP packets, the <3000-3699> range IP hardware access-list will match any ICMP packet that has the specified source and destination IP addresses and ICMP type.

You may apply the **any** parameter if the source or destination IP address is not important. The ICMP type is an optional parameter.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples Follow the below example commands to configure access-lists for ICMP, IP protocol and TCP.

ICMP Example To create an access-list that will permit ICMP packets with a source address of `192.168.1.0/24` with any destination address and an ICMP type of `5` enter the below commands:

```
awplus# configure terminal
awplus(config)# access-list 3000 permit icmp 192.168.1.0/24
any icmp-type 5
```

To destroy the access-list with an access-list identity of `3000` enter the below commands:

```
awplus# configure terminal
awplus(config)# no access-list 3000
```

IP Example To create an access-list that will permit any type of IP packet with a source address of `192.168.1.1` and any destination address, enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 3000 permit ip 192.168.1.1/32 any
```

To create an access-list that will deny all IGMP packets (IP protocol `2`) from the `192.168.0.0` network, enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 3000 deny proto 2 192.168.0.0/16
any
```

TCP Example To create an access-list that will permit TCP packets with a destination address of `192.168.1.1`, a destination port of 80 and any source address and source port, enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 3000 permit tcp any 192.168.1.1/32
eq 80
```

copy-to-mirror Example

To create an access-list that will copy-to-mirror TCP packets with a destination address of `192.168.1.1`, a destination port of 80 and any source address and source port for use with the **mirror interface** command, enter the commands:

```
awplus# configure terminal

awplus(config)# access-list 3000 copy-to-mirror tcp any
192.168.1.1/32 eq 80
```

Related Commands

access-group
mirror interface
show running-config
show access-list (IPv4 Hardware ACLs)

access-list (hardware MAC numbered)

This command creates an access-list for use with hardware classification, such as QOS. The access-list will match on packets that have the specified source and destination MAC addresses. The parameter **any** may be specified if an address does not matter.

The **no** variant of this command removes the specified MAC hardware filter access-list.

Syntax

```
access-list <4000-4699> {copy-to-cpu|copy-to-mirror|deny|permit|
    send-to-cpu} {<source-mac-address> <source-mac-mask>|
    any}{<destination-mac-address> <destination-mac-mask>|any}

no access-list <4000-4699>
```

Parameter	Description
<4000-4699>	Hardware MAC access-list.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`deny`	Access-list rejects packets that match the source and destination filtering.
`permit`	Access-list permits packets that match the source and destination filtering.
`send-to-cpu`	Specify packets to send to the CPU.
<source-mac-address>	The source MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> where each *H* is a hexadecimal number that represents a 4 bit binary number.
<source-mac-mask	The mask that will be applied to the source MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> where each *H* is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any source MAC address.
<destination-mac-address>	The destination MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number that represents a 4 bit binary number.
<destination-mac-mask>	The mask that will be applied to the destination MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any destination MAC address.
`vlan`	Specifies that the ACL will match on the ID in the packet's VLAN tag.
<1-4094>	The VLAN VID.

Parameter (cont.)	Description (cont.)
`inner-vlan`	This parameter is used within double-tagged VLANs. It is the inner VLAN tag (VID); sometimes referred to as the C-TAG (Customer VLAN TAG), where the vlan VID tag is referred to as the S-TAG (Service VLAN TAG).
<1-4094>	The inner VLAN VID.

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage This command creates an access-list for use with hardware classification, such as when applying QoS. The <4000-4699> range MAC hardware access-list will match on packets that have the specified source and destination MAC addresses. You may apply the **any** parameter if the source or destination MAC host address is not important.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To create an access-list that will permit packets with a MAC address of `0000.00ab.1234` and any destination address enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 4000 permit 0000.00ab.1234
0000.0000.0000 any
```

To create an access-list that will permit packets with an initial MAC address component of `0000.00ab` and any destination address, enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 4001 permit 0000.00ab.1234
0000.0000.FFFF any
```

To create an access-list that will copy-to-mirror packets with an initial MAC address component of `0000.00ab` and any destination address for use with the **mirror interface** command, enter the commands:

```
awplus# configure terminal
awplus(config)# access-list 4001 copy-to-mirror 0000.00ab.1234
0000.0000.FFFF any
```

To destroy the access-list with an access-list identity of `4000` enter the commands:

```
awplus# configure terminal
awplus(config)# no access-list 4000
```

Related Commands

access-group
mirror interface
show running-config
show access-list (IPv4 Hardware ACLs)

access-list hardware (named)

This command creates a named hardware access-list that can be applied to a switch port interface. ACL filters for a named hardware ACL are created in the IPv4 Hardware ACL Configuration mode.

The **no** variant of this command removes the specified named hardware ACL.

Syntax

```
access-list hardware <hardware-access-list-name>
no access-list hardware <hardware-access-list-name>
```

Parameter	Description
<hardware-access-list-name>	Specify the hardware ACL name to then define ACL filters for in the subsequent IPv4 Hardware ACL Configuration mode.

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage Use this command to name a hardware ACL and enter the IPv4 Hardware ACL Configuration mode. If the named hardware ACL doesn't exist, it will be created after entry. If the named hardware ACL does exist, then you can enter IPv4 Hardware ACL Configuration mode for that existing ACL.

Entering this command with the hardware ACL name moves you to the `(config-ip-hw-acl)` prompt for the IPv4 Hardware ACL Configuration mode so you can enter ACL filters with sequence numbers. From this prompt, configure the filters for the ACL. See **Chapter 57, Access Control Lists Introduction** for complete examples of configured sequenced numbered ACLs.

See also the table **"IPv4 Hardware Access List Commands and Prompts"** in this chapter. This table shows the relevant prompts at which ACL commands and ACL filters are entered for sequenced ACLs.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To create the hardware access-list named `ACL-1` and enter the IPv4 Hardware ACL Configuration mode to specify the ACL filter entry, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware ACL-1
awplus(config-ip-hw-acl)#
```

To remove the hardware access-list named `ACL-1`, use the commands:

```
awplus# configure terminal
awplus(config)# no access-list hardware ACL-1
```

Related Commands

access-group
(access-list hardware ICMP filter)
(access-list hardware IP protocol filter)
(access-list hardware TCP UDP filter)
(access-list standard named filter)
show access-list (IPv4 Hardware ACLs)

(access-list hardware ICMP filter)

Use this ACL filter to add a new ICMP filter entry to the current hardware access-list. The filter will match on any ICMP packet that has the specified source and destination IP addresses and ICMP type. The parameter **any** may be specified if an address does not matter and the ICMP type is an optional parameter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an ICMP filter entry from the current hardware access-list. You can specify the ICMP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its ICMP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the command, the **show access-list (IPv4 Hardware ACLs)** command on page 58.35.

Syntax [icmp]

```
[<sequence-number>] {deny|permit|send-to-cpu|copy-to-cpu|
   copy-to-mirror} icmp <source> <destination> [icmp <icmp-value>]

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror} icmp <source>
   <destination> [icmp <icmp-value>]

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`icmp`	ICMP packet type.

Parameter (cont.)	Description (cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> <reverse-mask>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	`any`	Matches any source IP address.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> <reverse-mask>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	`any`	Matches any destination IP address.
`icmp-type`	The ICMP type.	
<icmp-value>	The value of the ICMP type.	

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage

First create a named hardware access-list that applies the appropriate permit/deny requirements. Then use the **access-group** command on page 58.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note

You must reach the prompt `awplus(config-ip-hw-acl)#` by running the **access-list hardware (named)** command on page 58.18, and entering an appropriate access-list name.

Note

Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples

To add an access-list filter entry with a sequence number of `100` to the access-list named `my-list` that will permit ICMP packets with a source address of `192.168.1.0/24`, any destination address and an icmp type of `5`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# 100 permit icmp 192.168.1.0/24 any
icmp-type 5
```

To remove an access-list filter entry with a sequence number of `100` in the access-list named `my-list`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# no 100
```

Related Commands

access-list hardware (named)
show running-config
show access-list (IPv4 Hardware ACLs)

(access-list hardware IP protocol filter)

Use this ACL filter to add an IP protocol type filter entry to the current hardware access-list. The filter will match on any IP packet that has the specified source and destination IP addresses and IP protocol type, or has the optionally specified source and destination MAC addresses. The parameter **any** may be specified if an address does not matter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an IP protocol type filter entry from the current hardware access-list. You can specify the IP protocol type filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its IP protocol type filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Hardware ACLs)** command on page 58.35.

Syntax [any|ip|proto]

```
[<sequence-number>] {deny|permit|send-to-cpu|copy-to-cpu|
   copy-to-mirror} {any|ip|proto <ip-protocol>} {<source>|
   dhcpsnooping|any} {<destination>|any} [mac {<mac-source-address>
   <mac-source-mask>|any] {<mac-destination-address>
   <mac-destination-mask>|any}

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror} {any|ip|proto
   <ip-protocol>} {<source>|dhcpsnooping} {<destination>|any}
   [mac {<mac-source-address> <mac-source-mask>|any]
   {<mac-destination-address> <mac-destination-mask>|any}

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Access-list rejects packets of the type specified.
permit	Access-list allows packets of the type specified
send to cpu	Specify packets to send to the CPU.
copy to cpu	Specify packets to copy to the CPU.
copy to mirror	Specify packets to copy to the mirror port.
ip	IP packets.
any	Any packet.
proto *<ip-protocol>*	The IP Protocol type specified by it protocol number <1-255>.

Parameter (cont.)	Description (cont.)
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**)

Protocol Number	**Protocol Description [RFC Reference]**
1	Internet Control Message [RFC792]
2	Internet Group Management [RFC1112]
3	Gateway-to-Gateway [RFC823]
4	IP in IP [RFC2003]
5	Stream [RFC1190] [RFC1819]
6	TCP (Transmission Control Protocol) [RFC793]
8	EGP (Exterior Gateway Protocol) [RFC888]
9	IGP (Interior Gateway Protocol) [IANA]
11	Network Voice Protocol [RFC741]
17	UDP (User Datagram Protocol) [RFC768]
20	Host monitoring [RFC869]
27	RDP (Reliable Data Protocol) [RFC908]
28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
30	Bulk Data Transfer Protocol [RFC969]
33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
48	DSR (Dynamic Source Routing Protocol) [RFC4728]
50	ESP (Encap Security Payload) [RFC2406]
51	AH (Authentication Header) [RFC2402]

Parameter (cont.)	Description (cont.)	
<ip-protocol> (cont.)	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	58	ICMP for IPv6 [RFC1883]
	59	No Next Header for IPv6 [RFC1883]
	60	Destination Options for IPv6 [RFC1883]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	OSPFIGP [RFC1583]
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA
`dhcpsnooping`	The source address learned from the DHCP Snooping binding database.	

Parameter (cont.)	Description (cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/`*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/`*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
`mac`	Signifies a MAC and based hardware access-list.	
<mac-source-address>	The source host's MAC address, entered in HHHH.HHHH.HHHH format.	
<mac-source-mask>	The source host's MAC wildcard mask entered in HHHH.HHHH.HHHH format. where Hex `FF` = Ignore, and Hex `00` = Match.	

Parameter (cont.)	Description (cont.)
`any`	Matches any source MAC address.
`<mac-destination-address>`	The destination host's MAC address, entered in HHHH.HHHH.HHHH format.
`<mac-destination-mask>`	The destination host's wildcard mask entered in HHHH.HHHH.HHHH format. where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Matches any destination MAC address.

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit/deny requirements. Then use the **access-group** command on page 58.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note
The access control list being configured is selected by running the **access-list hardware (named)** command on page 58.18. with the required access control list number, or name, but with no further parameters selected.

Note
Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To add an access-list filter entry to the access-list named `my-list` that will permit any type of IP packet with a source address of `192.168.1.1` and any destination address, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# permit ip 192.168.1.1/32 any
```

To add an access-list filter entry to the access-list named `my-list` that will permit any type of IP packet with a source address of `192.168.1.1` and a MAC source address of `ffee.ddcc.bbaa` with any IP and MAC destination address, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# permit ip 192.168.1.1/32 any mac
ffee.ddcc.bbaa any
```

To add an access-list filter entry to the access-list named `my-list` a filter that will deny all IGMP packets (protocol `2`) from the `192.168.0.0` network with sequence number `50` in access-list, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# 50 deny proto 2 192.168.0.0/16 any
```

Related Commands

access-list hardware (named)
show running-config
show access-list (IPv4 Hardware ACLs)

(access-list hardware MAC filter)

Use this ACL filter to add a MAC filter entry to the current hardware access-list. The filter will match on any IP packet that has the specified source and destination MAC addresses. The parameter **any** may be specified if an address does not matter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes a MAC filter entry from the current hardware access-list. You can specify the MAC filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its MAC filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Hardware ACLs)** command on page 58.35.

Syntax [mac]

```
[<sequence-number>] {deny|permit|send-to-cpu|copy-to-cpu|
   copy-to-mirror} mac {<source-mac-address> <source-mac-mask>|any}
   {<destination-mac-address> <destination-mac-mask>|any}

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror}
   mac {<source-mac-address> <source-mac-mask>|any}
   {<destination-mac-address> <destination-mac-mask>|any}

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Specify packets to reject.

Parameter(cont.)	Description(cont.)
`permit`	Specify packets to accept.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the CPU.
`mac`	MAC address.
`<source-mac-address>`	The source MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number that represents a 4 bit binary number.
`<source-mac-mask>`	The mask that will be applied to the source MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any source MAC host.
`<destination-mac-address>`	The destination MAC address of the packets. Enter this in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number that represents a 4 bit binary number.
`<destination-mac-mask>`	The mask that will be applied to the destination MAC addresses. Enter this in the format <HHHH.HHHH.HHHH> where each H is a hexadecimal number that represents a 4 bit binary number. For a mask, each value will be either `0` or `F`. Where Hex `FF` = Ignore, and Hex `00` = Match.
`any`	Any destination MAC host.

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit/deny requirements. Then use the **access-group** command on page 58.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number

Note

The access control list being configured is selected by running the **access-list hardware (named)** command on page 58.18. with the required access control list number, or name, but with no further parameters selected.

Note

Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples

To add an access-list filter entry to the access-list named `my-list` that will permit packets with a source MAC address of `0000.00ab.1234` and any destination MAC address, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# permit mac 0000.00ab.1234
0000.0000.0000 any
```

To remove an access-list filter entry that permit packets with a source MAC address of `0000.00ab.1234` and any destination MAC address, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-list
awplus(config-ip-hw-acl)# no permit mac 0000.00ab.1234
0000.0000.0000 any
```

Related Commands

access-group
access-list hardware (named)
show running-config

(access-list hardware TCP UDP filter)

Use this ACL filter to add a TCP or UDP filter entry to the current hardware access-list. The filter will match on any TCP or UDP type packet that has the specified source and destination IP addresses. The parameter **any** may be specified if an address does not matter. If a sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes a TCP or UDP filter entry from the current hardware access-list. You can specify the TCP or UDP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its TCP or UDP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Hardware ACLs)** command on page 58.35.

Syntax [tcp|udp]

```
[<sequence-number>] {deny|permit|send-to-cpu|copy-to-cpu|
   copy-to-mirror} {tcp|udp} [<source>|eq <sourceport>|
   gt <sourceport>|lt <sourceport>|ne <sourceport>|
   range <start-range> <end-range>] [<destination>|eq <destport>|
   gt <destport>|lt <destport>|ne <destport>|
   range <start-range> <end-range>]

no {deny|permit|send-to-cpu|copy-to-cpu|copy-to-mirror} {tcp|udp}
   [<source>|eq <sourceport>|gt <sourceport>|lt <sourceport>|
   ne <sourceport>|range <start-range> <end-range>] [<destination>|
   eq <destport>|gt <destport>|lt <destport>|ne <destport>|
   range <start-range> <end-range>]

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Access-list rejects packets that match the source and destination filtering specified with this command.
permit	Access-list permits packets that match the source and destination filtering specified with this command.
send-to-cpu	Specify packets to send to the CPU.
copy-to-cpu	Specify packets to copy to the CPU.
copy-to-mirror	Specify packets to copy to the mirror port.
tcp	TCP packets.
udp	UDP packets.

Parameter(cont.)	Description(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/`*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
<sourceport>	The source TCP or UDP port number, specified as an integer between 0 and 65535.	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/`*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
`eq`	Equal to.	
`lt`	Less than.	
`gt`	Greater than.	

Parameter(cont.)	Description(cont.)
`ne`	Not equal to.
<destport>	The source TCP or UDP port number, specified as an integer between 0 and 65535.
`range`	Specify the range of port numbers between 0 and 65535.
<start-range>	The source or destination port number at the start of the range *<0-65535>*.
<end-range>	The source or destination port number at the end of the range *<0-65535>*.

Mode IPv4 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage First create a named hardware access-list that applies the appropriate permit/deny requirements. Then use the **access-group** command on page 58.4 to apply this access-list to a specific port or range. Note that this command will apply the access-list only to **incoming** data packets.

An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note
The access control list being configured is selected by running the **access-list hardware (named)** command on page 58.18. with the required access control list number, or name, but with no further parameters selected.

Note
Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Example To add an access-list filter entry to access-list named `my-hw-list` that will permit TCP packets with a destination address of `192.168.1.1`, a destination port of `80`, and any source address, and source port, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware my-hw-list
awplus(config-ip-hw-acl)# permit tcp any 192.168.1.1/32 eq 80
```

Related Commands
access-list hardware (named)
show running-config
show access-list (IPv4 Hardware ACLs)

commit (IPv4)

Use this command to commit the IPv4 ACL filter configuration entered at the console to the hardware immediately without exiting the IPv4 Hardware ACL Configuration mode.

This command forces the associated hardware and software IPv4 ACLs to synchronize.

Syntax `commit`

Mode IPv4 Hardware ACL Configuration

Usage Normally, when an IPv4 hardware ACL is edited, the new configuration state of the IPv4 ACL is not written to hardware until you exit IPv4 Hardware ACL Configuration mode. By entering this command you can ensure that the current state of a hardware access-list that is being edited is written to hardware immediately.

Scripts typically do not include the **exit** command to exit configuration modes, potentially leading to IPv4 ACL filters in hardware not being correctly updated. Using this **commit** command in a configuration script after specifying an IPv4 hardware ACL filter ensures that it is updated in the hardware immediately.

Example To update the hardware with the IPv4 ACL filter configuration, use the command:

```
awplus# configure terminal
awplus(config)# access-list hardware my-hw-list
awplus(config-ip-hw-acl)# commit
```

Related Commands **access-list hardware (named)**

show access-list (IPv4 Hardware ACLs)

Use this command to display the specified access-list, or all access-lists if none have been specified. Note that only defined access-lists are displayed. An error message is displayed for an undefined access-list.

Syntax

```
show access-list [<1-99>|<100-199>|<1300-1999>|<2000-2699>|
    <3000-3699>|<4000-4499>|<access-list-name>]
```

Parameter	Description
<1-99>	IP standard access-list.
<100-199>	IP extended access-list.
<1300-1999>	IP standard access-list (standard - expanded range).
<2000-2699>	IP extended access-list (extended - expanded range).
<3000-3699>	Hardware IP access-list.
<4000-4499>	Hardware MAC access-list.
<access-list-name>	IP named access-list.

Mode User Exec and Privileged Exec

Examples To show all access-lists configured on the switch:

```
awplus# show access-list
```

```
Standard IP access list 1
    deny 172.16.2.0, wildcard bits 0.0.0.255
Standard IP access list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
Hardware IP access list 3001
    permit ip 192.168.20.0 255.255.255.0 any
Hardware IP access list 3020
    permit tcp any 192.0.2.0/24
awplus#show access-list 20
```

To show the access-list with an ID of 20:

```
awplus# show access-list 20
```

```
Standard IP access-list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
```

Note the below error message if you attempt to show an undefined access-list:

```
awplus# show access-list 2
```

```
% Can't find access-list 2
```

Related Commands

access-list extended (named)
access-list (hardware MAC numbered)
access-list hardware (named)

show interface access-group

Use this command to display the access groups attached to a port. If an access group is specified, then the output only includes the ports that the specified access group is attached to. If no access group is specified then this command displays all access groups that are attached to the ports that are specified with *<port-list>*.

Note that **access group** is the term given for an access-list when it is applied to an interface.

Note This command will function on your switch in stand-alone mode. but is not supported when the device forms part of a VCStack.

Syntax

```
show interface <port-list> access-group [<3000-3699>|<4000-4699>]
```

Parameter	Description
<port-list>	Specify the ports to display information. A port-list can be either: ■ a switch port (e.g. port1.0.6) a static channel group (e.g. sa2) or a dynamic (LACP) channel group (e.g. po2) ■ a continuous range of ports separated by a hyphen, e.g. port1.0.1-1.0.6 or port1.0.1-port1.0.6 or po1-po2 ■ a comma-separated list of ports and port ranges, e.g. port1.0.1,port1.0.3-1.0.6. Do not mix switch ports, static channel groups, and LACP channel groups in the same list.
`access group`	Select the access group whose details you want to show.
<3000-3699>	Specifies the Hardware IP access-list.
<4000-4699>	Specifies the Hardware MAC access-list.

Mode User Exec and Privileged Exec

Example To show all access-lists attached to `port1.0.1`, use the command:

```
awplus# show interface port1.0.1 access-group
```

Output **Figure 58-1: Example output from the show interface access-group command**

```
Interface port1.0.1
  access-group 3000
  access-group 3002
  access-group 3001
```

Related Commands access-group

Chapter 59: IPv4 Software Access Control List (ACL) Commands

Introduction 59.2
IPv4 Software Access List Commands and Prompts 59.3
Command List 59.4
access-list extended (named) 59.4
access-list (extended numbered) 59.12
(access-list extended ICMP filter) 59.15
(access-list extended IP filter) 59.17
(access-list extended IP protocol filter) 59.20
(access-list extended TCP UDP filter) 59.24
access-list standard (named) 59.27
access-list (standard numbered) 59.29
(access-list standard named filter) 59.31
(access-list standard numbered filter) 59.33
clear ip prefix-list 59.35
dos 59.36
ip prefix-list 59.40
maximum-access-list 59.42
show access-list (IPv4 Software ACLs) 59.43
show dos interface 59.45
show ip access-list 59.47
show ip prefix-list 59.48

Introduction

This chapter provides an alphabetical reference for the IPv4 Software Access Control List (ACL) commands, and contains detailed command information and command examples about IPv4 software ACLs as applied to Routing and Multicasting, which are not applied to interfaces.

Note

See **Chapter 57, Access Control Lists Introduction** for descriptions of ACLs, and for further information about rules when applying ACLs see the **ACL Rules** section.

See **ACL Filter Sequence Numbers** and **ACL Filter Sequence Number Behavior** sections in **Chapter 57, Access Control Lists Introduction** about ACL Filters.

See all relevant Routing commands and configurations in **"IPv4 Software Access Control List (ACL) Commands"** and all relevant Multicast commands and configurations in **"Multicast Applications"**.

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself. For more information on link aggregation see **Chapter 22, Link Aggregation Introduction and Configuration**, and **Chapter 23, Link Aggregation Commands**.

Note

Text in parenthesis in command names indicates usage not keyword entry. For example, **access-list hardware (named)** indicates named IPv4 hardware ACLs entered as `access-list hardware` <*name*> where <*name*> is a placeholder not a keyword.

Note

Parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI, such as **(access-list standard numbered filter)** represents command entry in the format shown in the syntax `[<sequence-number>] {deny|permit} {<source>|host <host-address>|any}`.

Note

Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

IPv4 Software Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table **"IPv4 Software Access List Commands and Prompts"** shows the CLI prompts at which ACL commands are entered.

Table 59-1: IPv4 Software Access List Commands and Prompts

Command Name	Command Mode	Prompt
clear ip prefix-list	Privileged Exec	`awplus#`
show ip access-list	Privileged Exec	`awplus#`
show ip prefix-list	Privileged Exec	`awplus#`
access-group	Global Configuration	`awplus(config)#`
access-list (extended numbered)	Global Configuration	`awplus(config)#`
access-list standard (named)	Global Configuration	`awplus(config)#`
access-list (standard numbered)	Global Configuration	`awplus(config)#`
ip prefix-list	Global Configuration	`awplus(config)#`
maximum-access-list	Global Configuration	`awplus(config)#`
dos	Interface Configuration	`awplus(config-if)#`
(access-list extended ICMP filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list extended IP filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list extended IP protocol filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list extended TCP UDP filter)	IPv4 Extended ACL Configuration	`awplus(config-ip-ext-acl)#`
(access-list standard named filter)	IPv4 Standard ACL Configuration	`awplus(config-ip-std-acl)#`
(access-list standard numbered filter)	IPv4 Standard ACL Configuration	`awplus(config-ip-std-acl)#`

Command List

access-list extended (named)

This command configures an extended named access-list that permits or denies packets from specific source and destination IP addresses. You can either create an extended named ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Extended ACL Configuration mode for sequenced ACL filter entry after entering a list name.

The **no** variant of this command removes a specified extended named access-list.

Syntax [list-name]

```
access-list extended <list-name>

no access-list extended <list-name>
```

Parameter	Description
<list-name>	A user-defined name for the access-list

Syntax [icmp]

```
access-list extended <list-name>{deny|permit}
   icmp <source> <destination> [icmp-type <type-number>] [log]

no access-list extended <list-name>{deny|permit}
   icmp <source> <destination> [icmp-type <type-number>] [log]
```

Table 59-2: Parameters in the access-list extended (named) command - icmp

Parameter	Description
<list-name>	A user-defined name for the access-list.
`deny`	The access-list rejects packets that match the type, source, and destination filtering specified with this command.
`permit`	The access-list permits packets that match the type, source, and destination filtering specified with this command.
`icmp`	The access-list matches only ICMP packets.
`icmp-type`	Matches only a specified type of ICMP messages. This is valid only when the filtering is set to match ICMP packets.

Table 59-2: Parameters in the access-list extended (named) command - icmp (cont.)

Parameter(cont.)	Description(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

Table 59-2: Parameters in the access-list extended (named) command - icmp (cont.)

Parameter(cont.)	Description(cont.)	
<type-number>	The ICMP type, as defined in RFC792 and RFC950. Specify one of the following integers to create a filter for the ICMP message type:	
	0	Echo replies.
	3	Destination unreachable messages.
	4	Source quench messages.
	5	Redirect (change route) messages.
	8	Echo requests.
	11	Time exceeded messages.
	12	Parameter problem messages.
	13	Timestamp requests.
	14	Timestamp replies.
	15	Information requests.
	16	Information replies.
	17	Address mask requests.
	18	Address mask replies.
log	Logs the results.	

Syntax [tcp|udp]

```
access-list extended <list-name> {deny|permit} {tcp|udp} <source>
   [eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>]
   <destination> [eq <destport>|lt <destport>|gt <destport>|
   ne <destport>] [log]

no access-list extended <list-name> {deny|permit} {tcp|udp} <source>
   [eq <sourceport>|lt <sourceport>|gt <sourceport>|ne <sourceport>]
   <destination> [eq <destport> |lt <destport>|gt <destport>|ne
   <destport>][log]
```

Table 59-3: Parameters in the access-list extended (named) command - tcp|udp

Parameter	Description
<list-name>	A user-defined name for the access-list.
deny	The access-list rejects packets that match the type, source, and destination filtering specified with this command.
permit	The access-list permits packets that match the type, source, and destination filtering specified with this command.
tcp	The access-list matches only TCP packets.
udp	The access-list matches only UDP packets.

Table 59-3: Parameters in the access-list extended (named) command - tcp|udp

Parameter(cont.)	Description(cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/` *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>`/` *<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
<sourceport>	The source port number, specified as an integer between 0 and 65535.	
<destport>	The destination port number, specified as an integer between 0 and 65535.	
`eq`	Matches port numbers equal to the port number specified immediately after this parameter.	
`lt`	Matches port numbers less than the port number specified immediately after this parameter.	
`gt`	Matches port numbers greater than the port number specified immediately after this parameter.	

Table 59-3: Parameters in the access-list extended (named) command - tcp|udp

Parameter(cont.)	Description(cont.)
`ne`	Matches port numbers not equal to the port number specified immediately after this parameter.
`log`	Log the results.

Syntax [proto|any|ip]

```
access-list extended <list-name> {deny|permit} {proto <ip-protocol>|
   any|ip} {<source>} {<destination>} [log]

no access-list extended <list-name>{deny|permit}
   {proto <ip-protocol>|any|ip}{<source>}{<destination>}[log]
```

Table 59-4: Parameters in the access-list extended (named) command - proto|ip|any

Parameter	Description	
<list-name>	A user-defined name for the access-list.	
`deny`	The access-list rejects packets that match the type, source, and destination filtering specified with this command.	
`permit`	The access-list permits packets that match the type, source, and destination filtering specified with this command.	
`proto`	Matches only a specified type of IP Protocol.	
`any`	The access-list matches any type of IP packet.	
`ip`	The access-list matches only IP packets.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	<ip-addr> <reverse-mask>	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.

Table 59-4: Parameters in the access-list extended (named) command - proto|ip|any

Parameter (cont.)	Description (cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr>/*<prefix>*	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr> *<reverse-mask>*	Alternatively, you can enter a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
`log`	Logs the results.	
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**)	
	Protocol Number	**Protocol Description [RFC Reference]**
	1	Internet Control Message [RFC792]
	2	Internet Group Management [RFC1112]
	3	Gateway-to-Gateway [RFC823]
	4	IP in IP [RFC2003]
	5	Stream [RFC1190] [RFC1819]
	6	TCP (Transmission Control Protocol) [RFC793]
	8	EGP (Exterior Gateway Protocol) [RFC888]
	9	IGP (Interior Gateway Protocol) [IANA]
	11	Network Voice Protocol [RFC741]
	17	UDP (User Datagram Protocol) [RFC768]
	20	Host monitoring [RFC869]
	27	RDP (Reliable Data Protocol) [RFC908]
	28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
	29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
	30	Bulk Data Transfer Protocol [RFC969]

Table 59-4: Parameters in the access-list extended (named) command - proto|ip|any

Parameter (cont.)	Description (cont.)	
<ip-protocol> (cont.)	**Protocol Number**	**Protocol Description [RFC Reference]**
	33	Datagram Congestion Control Protocol [RFC4340]
	48	DSR (Dynamic Source Routing Protocol) [RFC4728]
	50	ESP (Encap Security Payload) [RFC2406]
	51	AH (Authentication Header) [RFC2402]
	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	OSPFIGP [RFC1583]
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring access-list for filtering IP software packets. To enable backwards compatibility you can either create access-lists from within this command, or you can enter **access-list** followed by only the number. This latter method moves you to the IPv4 Extended ACL Configuration mode for the selected access-list number, and from here you can configure your access-lists by using the commands **(access-list extended ICMP filter)**, **(access-list extended IP filter)**, and **(access-list extended IP protocol filter)**.

The table **"IPv4 Software Access List Commands and Prompts" on page 59.3** shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note that packets must match both the source and the destination details.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples You can enter the extended named ACL in the Global Configuration mode together with the ACL filter entry on the same line, as shown below:

```
awplus# configure terminal
awplus(config)# access-list extended TK deny tcp 2.2.2.3/24 eq
14 3.3.3.4/24 eq 12 log
```

Alternatively, you can enter the extended named ACL in Global Configuration mode before specifying the ACL filter entry in the IPv4 Extended ACL Configuration mode, as shown below:

```
awplus# configure terminal
awplus(config)# access-list extended TK
awplus(config-ip-ext-acl)# deny tcp 2.2.2.3/24 eq 14 3.3.3.4/24
eq 12 log
```

Related Commands
(access-list extended ICMP filter)
(access-list extended IP filter)
(access-list extended TCP UDP filter)
show running-config
show ip access-list

access-list (extended numbered)

This command configures an extended numbered access-list that permits or denies packets from specific source and destination IP addresses. You can either create an extended numbered ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Extended ACL Configuration mode for sequenced ACL filter entry after entering a list number.

The **no** variant of this command removes a specified extended named access-list.

Syntax [list-number]

```
access-list {<100-199>|<2000-2699>}

no access-list {<100-199>|<2000-2699>}
```

Parameter	Description
<100-199>	IP extended access-list.
<2000-2699>	IP extended access-list (expanded range).

Syntax [deny|permit]

```
access-list {<100-199>|<2000-2699>} {deny|permit} ip <source>
   <destination>

no access-list {<100-199>|<2000-2699>}{deny|permit} ip <source>
   <destination>
```

Parameter	Description	
<100-199>	IP extended access-list.	
<2000-2699>	IP extended access-list (expanded range).	
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.	
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr> *<reverse-mask>*	An IPv4 address, followed by a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`. This matches any source IP address within the specified subnet.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	

Parameter (cont.)	Description (cont.)	
	`any`	Matches any destination IP address.
	`host <ip-addr>`	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr> *<reverse-mask>*	An IPv4 address, followed by a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`. This matches any destination IP address within the specified subnet.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring access-list for filtering IP software packets. To enable backwards compatibility you can either create access-lists from within this command, or you can enter **access-list** followed by only the number. This latter method moves you to the IPv4 Extended ACL Configuration mode for the selected access-list number, and from here you can configure your access-lists by using the commands **(access-list extended ICMP filter)**, **(access-list extended IP filter)**, and **(access-list extended IP protocol filter)**.

The table **"IPv4 Software Access List Commands and Prompts" on page 59.3** shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note that packets must match both the source and the destination details.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples You can enter the extended named ACL in the Global Configuration mode together with the ACL filter entry on the same line, as in previous software releases as shown below:

```
awplus# configure terminal
awplus(config)# access-list 101 deny ip 172.16.10.0 0.0.0.255
any
```

Alternatively, you can enter the extended named ACL in Global Configuration mode before specifying the ACL filter entry in the IPv4 Extended ACL Configuration mode, as shown below:

```
awplus# configure terminal
awplus(config)# access-list 101
awplus(config-ip-ext-acl)# deny ip 172.16.10.0 0.0.0.255 any
```

Related Commands

(access-list extended ICMP filter)
(access-list extended IP filter)
(access-list extended TCP UDP filter)
show running-config
show ip access-list

(access-list extended ICMP filter)

Use this ACL filter to add a new ICMP filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an ICMP filter entry from the current extended access-list. You can specify the ICMP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its ICMP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Software ACLs)** command.

Syntax [icmp]

```
[<sequence-number>] {deny|permit} icmp <source> <destination>
    [icmp-type <icmp-value>] [log]

no {deny|permit} icmp <source> <destination>[icmp-type <icmp-value>]
    [log]

no <sequence-number>
```

Parameter	Description	
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.	
deny	Access-list rejects packets that match the source and destination filtering specified with this command.	
permit	Access-list permits packets that match the source and destination filtering specified with this command.	
icmp	ICMP packet type.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	any	Matches any source IP address.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	any	Matches any destination IP address.
icmp-type	The ICMP type.	

Parameter (cont.)	Description (cont.)
<icmp-value>	The value of the ICMP type.
log	Log the results.

Mode IPv4 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note
The access control list being configured is selected by running the **access-list (extended numbered)** command or the **access-list extended (named)** command, with the required access control list number, or name - but with no further parameters selected.

Note
Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add a new entry in access-list called `my-list` that will reject ICMP packets from `10.0.0.1` to `192.168.1.1`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# deny icmp 10.0.0.1/32 192.168.1.1/32
```

Use the following commands to add a new filter at sequence number `5` position of the access-list called `my-list`. The filter will accept the ICMP type `8` packets from `10.1.1.0/24` network, to `192.168.1.0` network:

```
awplus# configure terminal
awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# 5 permit icmp 10.1.1.0/24
192.168.1.0/24 icmp-type 8
```

Related Commands **access-group**
show running-config
show ip access-list

(access-list extended IP filter)

Use this ACL filter to add a new IP filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an IP filter entry from the current extended access-list. You can specify the IP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its IP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Software ACLs)** command.

Syntax [ip]

```
[<sequence-number>] {deny|permit} ip <source> <destination>
no {deny|permit} ip <source> <destination>
no <sequence-number>
```

Parameter	Description	
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.	
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.	
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	`any`	Matches any source IP address.
	`host` *<ip-addr>*	Matches a single source host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr> *<reverse-mask>*	Alternatively, enter an IPv4 address followed by a reverse mask in dotted decimal format. For example, enter `192.168.1.1 0.0.0.255`.

Parameter (cont.)	Description (cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	`any`	Matches any destination IP address.
	`host` *<ip-addr>*	Matches a single destination host with the IP address given by *<ip-addr>* in dotted decimal notation.
	<ip-addr> *<reverse-mask>*	Alternatively, enter an IPv4 address followed by a reverse mask in dotted decimal format. For example, enter `192.168.1.1 0.0.0.255`.

Mode Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note

The access control list being configured is selected by running the **access-list (extended numbered)** command or the **access-list extended (named)** command, with the required access control list number, or name - but with no further parameters selected.

Note

Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [list-number]

First use the following commands to enter the IPv4 Extended ACL Configuration mode and define a numbered extended access-list `101`:

```
awplus# configure terminal
awplus(config)# access-list 101
awplus(config-ip-ext-acl)#
```

Then use the following commands to add a new entry to the numbered extended access-list `101` that will reject packets from `10.0.0.1` to `192.168.1.1`:

```
awplus(config-ip-ext-acl)# deny ip host 10.0.0.1 host
                           192.168.1.1
awplus(config-ip-ext-acl)# 20 permit ip any any
```

Example 2 [list-name]

First use the following commands to enter the IPv4 Extended ACL Configuration mode and define a named access-list called `my-acl`:

```
awplus# configure terminal
awplus(config)# access-list extended my-acl
awplus(config-ip-ext-acl)#
```

Then use the following commands to add a new entry to the named access-list `my-acl` that will reject packets from `10.0.0.1` to `192.168.1.1`:

```
awplus(config-ip-ext-acl)# deny ip host 10.0.0.1 host
                           192.168.1.1
awplus(config-ip-ext-acl)# 20 permit ip any any
```

Example 3 [list-number]

Use the following commands to remove the access-list filter entry with sequence number `20` from extended numbered access-list `101`.

```
awplus# configure terminal
awplus(config)# access-list 101
awplus(config-ip-ext-acl)# no 20
```

Example 4 [list-name]

Use the following commands to remove the access-list filter entry with sequence number `20` from extended named access-list `my-acl`:

```
awplus# configure terminal
awplus(config)# access-list extended my-acl
awplus(config-ip-ext-acl)# no 20
```

Related Commands

access-list extended (named)
access-list (extended numbered)
show running-config
show ip access-list

(access-list extended IP protocol filter)

Use this ACL filter to add a new IP protocol type filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes an IP protocol filter entry from the current extended access-list. You can specify the IP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its IP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Software ACLs)** command.

Syntax [proto]

```
[<sequence-number>] {deny|permit} proto <ip-protocol> <source>
   <destination> [log]

no {deny|permit} proto <ip-protocol> <source> <destination> [log]

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
`deny`	Access-list rejects packets that match the source and destination filtering specified with this command.
`permit`	Access-list permits packets that match the source and destination filtering specified with this command.
`proto` *<ip-protocol>*	The IP Protocol type specified by its protocol number <1-255>.
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**).

Protocol Number	Protocol Description [RFC Reference]
1	Internet Control Message [RFC792]
2	Internet Group Management [RFC1112]
3	Gateway-to-Gateway [RFC823]
4	IP in IP [RFC2003]
5	Stream [RFC1190] [RFC1819]
6	TCP (Transmission Control Protocol) [RFC793]
8	EGP (Exterior Gateway Protocol) [RFC888]
9	IGP (Interior Gateway Protocol) [IANA]
11	Network Voice Protocol [RFC741]

Parameter (cont.)	Description (cont.)	
<ip-protocol> (cont.)	17	UDP (User Datagram Protocol) [RFC768]
	20	Host monitoring [RFC869]
	27	RDP (Reliable Data Protocol) [RFC908]
	28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
	29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
	30	Bulk Data Transfer Protocol [RFC969]
	33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
	48	DSR (Dynamic Source Routing Protocol) [RFC4728]
	50	ESP (Encap Security Payload) [RFC2406]
	51	AH (Authentication Header) [RFC2402]
	54	NARP (NBMA Address Resolution Protocol) [RFC1735]
	88	EIGRP (Enhanced Interior Gateway Routing Protocol)
	89	OSPFIGP [RFC1583]
	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA

Parameter (cont.)	Description (cont.)	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	any	Matches any source IP address.
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	any	Matches any destination IP address.
log	Log the results.	

Mode IPv4 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note The access control list being configured is selected by running the **access-list (extended numbered)** command or the **access-list extended (named)** command, with the required access control list number, or name - but with no further parameters selected.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [creating a list] Use the following commands to add a new access-list filter entry to the access-list named `my-list` that will reject IP packets from source address `10.10.1.1/32` to destination address `192.68.1.1/32`:

```
awplus# configure terminal
awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# deny ip 10.10.1.1/32 192.168.1.1/32
```

Example 2 [adding to a list]

Use the following commands to add a new access-list filter entry at sequence position `5` in the access-list named `my-list` that will accept packets from source address `10.10.1.1/24` to destination address `192.68.1.1/24`:

```
awplus# configure terminal
awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# 5 permit ip 10.10.1.1/24 192.168.1.1/
24
```

Related Commands

access-list extended (named)
access-list (extended numbered)
show running-config
show ip access-list

(access-list extended TCP UDP filter)

Use this ACL filter to add a new TCP or UDP filter entry to the current extended access-list. If the sequence number is specified, the new filter is inserted at the specified location. Otherwise, the new filter is added at the end of the access-list.

The **no** variant of this command removes a TCP or UDP filter entry from the current extended access-list. You can specify the TCP or UDP filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its TCP or UDP filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Software ACLs)** command.

Syntax [tcp|udp]

```
[<sequence-number>] {deny|permit} {tcp|udp} <source> {eq <sourceport>
   |lt <sourceport>|gt <sourceport>|ne <sourceport>} <destination>
   [eq <destport>|lt <destport>|gt <destport>|ne <destport>] [log]

no [<sequence-number>]{deny|permit} {tcp|udp} <source> {eq
   <sourceport> |lt <sourceport>|gt <sourceport>|ne
   <sourceport>}<destination> [eq <destport>|lt <destport>|gt
   <destport>|ne <destport>] [log]

no <sequence-number>
```

Parameter	Description	
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.	
deny	Access-list rejects packets that match the source and destination filtering specified with this command.	
permit	Access-list permits packets that match the source and destination filtering specified with this command.	
tcp	The access-list matches only TCP packets.	
udp	The access-list matches only UDP packets.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	any	Matches any source IP address.
<sourceport>	The source port number, specified as an integer between 0 and 65535.	

Parameter (cont.)	Description (cont.)	
<destination>	The destination address of the packets. You can specify a single host, a subnet, or all destinations. The following are the valid formats for specifying the destination:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	`any`	Matches any destination IP address.
<destport>	The destination port number, specified as an integer between 0 and 65535.	
`eq`	Matches port numbers equal to the port number specified immediately after this parameter.	
`lt`	Matches port numbers less than the port number specified immediately after this parameter.	
`gt`	Matches port numbers greater than the port number specified immediately after this parameter.	
`ne`	Matches port numbers not equal to the port number specified immediately after this parameter.	
`log`	Log the results.	

Mode IPv4 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note
The access control list being configured is selected by running the **access-list (extended numbered)** command or the **access-list extended (named)** command, with the required access control list number, or name - but with no further parameters selected.

Note
Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [creating a list] To add a new entry to the access-list named `my-list` that will reject TCP packets from `10.0.0.1` on TCP port `10` to `192.168.1.1` on TCP port `20`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# deny tcp 10.0.0.1/32 eq 10
                           192.168.1.1/32 eq 20
```

Example 2 [adding to a list]

To insert a new entry with sequence number `5` of the access-list named `my-list` that will accept UDP packets from `10.1.1.0/24` network to `192.168.1.0/24` network on UDP port `80`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list extended my-list
awplus(config-ip-ext-acl)# 5 permit udp 10.1.1.0/24
192.168.1.0/24 eq 80
```

Related Commands

access-list extended (named)
access-list (extended numbered)
show running-config
show ip access-list

access-list standard (named)

This command configures a standard named access-list that permits or denies packets from a specific source IP address. You can either create a standard named ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Standard ACL Configuration mode for sequenced ACL filter entry after first entering an access-list name.

The **no** variant of this command removes a specified standard named access-list.

Syntax [list-name]

```
access-list standard <standard-access-list-name>

no access-list standard <standard-access-list-name>
```

Parameter	Description
<standard-access-list-name>	Specify a name for the standard access-list.

Syntax [deny|permit]

```
access-list standard <standard-access-list-name> {deny|permit}
    <source>

no access-list standard <standard-access-list-name> {deny|permit}
    <source>
```

Parameter	Description	
<standard-access-list-name>	Specify a name for the standard access-list.	
deny	The access-list rejects packets that match the source filtering specified with this command.	
permit	The access-list permits packets that match the source filtering specified with this command.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any source IP address within the specified subnet.
	any	Matches any source IP address.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring a standard named access-list for filtering IP software packets. For backwards compatibility you can either create the access-list from within this command, or you can enter this command followed by only the standard access-list name then enter. This latter method moves you to the IPv4 Standard ACL Configuration mode for the selected standard named access-list, and from here you can configure the deny or permit filters for this selected standard named access-list.

See the table **"IPv4 Software Access List Commands and Prompts"** in this chapter which shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To define a standard access-list named `my-list` and deny any packets from any source, use the commands:

```
awplus# configure terminal
awplus(config)# access-list standard my-list deny any
```

Alternatively, to define a standard access-list named `my-list` and enter the IPv4 Standard ACL Configuration mode to deny any packets from any source, use the commands:

```
awplus# configure terminal
awplus(config)# access-list standard my-list
awplus(config-ip-std-acl)# 5 deny any
```

Related Commands
(access-list standard named filter)
show running-config
show ip access-list

access-list (standard numbered)

This command configures a standard numbered access-list that permits or denies packets from a specific source IP address. You can either create a standard numbered ACL together with an ACL filter entry in the Global Configuration mode, or you can use the IPv4 Standard ACL Configuration mode for sequenced ACL filter entry after first entering an access-list number.

The **no** variant of this command removes a specified standard numbered access-list.

Syntax [list-number]

```
access-list {<1-99>|<1300-1999>}
no access-list {<1-99>|<1300-1999>}
```

Parameter	Description
<1-99>	IP standard access-list.
<1300-1999>	IP standard access-list (expanded range).

Syntax [deny|permit]

```
access-list {<1-99>|<1300-1999>} {deny|permit} <source>
no access-list {<1-99>|<1300-1999>} {deny|permit} <source>
```

Parameter	Description	
<1-99>	IP standard access-list.	
<1300-1999>	IP standard access-list (expanded range).	
`deny`	Access-list rejects packets from the specified source.	
`permit`	Access-list accepts packets from the specified source.	
<source>	The source address of the packets. You can specify a single host, a subnet, or all sources. The following are the valid formats for specifying the source:	
	<ip-addr> <reverse-mask>	Enter an IPv4 address followed by a reverse mask in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
	`any`	Matches any source IP address.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use this command when configuring a standard numbered access-list for filtering IP software packets. For backwards compatibility you can either create the access-list from within this command, or you can enter this command followed by only the standard access-list name. This moves you to the IPv4 Standard ACL Configuration mode for the selected standard numbered access-list, and from here you can configure the deny or permit filters for this selected standard numbered access-list.

See the table **"IPv4 Software Access List Commands and Prompts"** in this chapter which shows the prompts at which ACL commands are entered. See the relevant links shown for the **Related Commands**.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To create ACL number `67` that will deny packets from subnet `172.16.10`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list 67 deny 172.16.10.0 0.0.0.255
```

Alternatively, to enter the IPv4 Standard ACL Configuration mode to create the ACL filter and deny packets from subnet `172.16.10.0` for the standard numbered access-list `67`, use the commands:

```
awplus# configure terminal
awplus(config)# access-list 67
awplus(config-ip-std-acl)# deny 172.16.10.0 0.0.0.255
```

Related Commands
(access-list standard named filter)
show running-config
show ip access-list

(access-list standard named filter)

This ACL filter adds a source IP address filter entry to a current named standard access-list. If the sequence number is specified, the new filter entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a source IP address filter entry from the current named standard access-list. You can specify the source IP address filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its source IP address filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Software ACLs)** command.

Syntax

```
[<sequence-number>] {deny|permit} {<source> [exact-match]|any}
no {deny|permit} {<source> [exact-match]|any}
no <sequence-number>
```

Parameter	Description	
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.	
deny	Access-list rejects packets of the source filtering specified.	
permit	Access-list allows packets of the source filtering specified	
<source>	The source address of the packets. You can specify either a subnet or all sources. The following are the valid formats for specifying the source:	
	<ip-addr>/<prefix>	An IPv4 address, followed by a forward slash, then the prefix length. This matches any destination IP address within the specified subnet.
	<ip-addr>	An IPv4 address in a.b.c.d format.
exact-match	Specify an exact IP prefix to match on.	
any	Matches any source IP address.	

Mode IPv4 Standard ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note

The access control list being configured is selected by running the **access-list standard (named)** command with the required access control list number, or name, but with no further parameters selected.

Note

Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples Use the following commands to add a new filter entry to access-list `my-list` that will reject IP address `10.1.1.1`:

```
awplus# configure terminal
awplus(config)# access-list standard my-list
awplus(config-ip-std-acl)# deny 10.1.1.1/32
```

Use the following commands to insert a new filter entry into access-list `my-list` at sequence position number `15` that will accept IP network `10.1.2.0`:

```
awplus# configure terminal
awplus(config)# access-list standard my-list
awplus(config-ip-std-acl)# 15 permit 10.1.2.0/24
```

Related Commands

access-list standard (named)
show running-config
show ip access-list

(access-list standard numbered filter)

This ACL filter adds a source IP address filter entry to a current standard numbered access-list. If a sequence number is specified, the new filter entry is inserted at the specified location. Otherwise, the new filter entry is added at the end of the access-list.

The **no** variant of this command removes a source IP address filter entry from the current standard numbered access-list. You can specify the source IP address filter entry for removal by entering either its sequence number (e.g. `no 10`), or by entering its source IP address filter profile without specifying its sequence number.

Note that the sequence number can be found by running the **show access-list (IPv4 Software ACLs)** command.

Syntax

```
[<sequence-number>] {deny|permit} {<source>|host <host-address>|any}
no {deny|permit} {<source>|host <host-address>|any}
no <sequence-number>
```

Parameter	Description	
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.	
deny	Access-list rejects packets of the type specified.	
permit	Access-list allows packets of the type specified	
<source>	The source address of the packets. You can specify either a subnet or all sources. The following are the valid formats for specifying the source:	
	<ip-addr> <reverse-mask>	Enter a reverse mask for the source address in dotted decimal format. For example, entering `192.168.1.1 0.0.0.255` is the same as entering `192.168.1.1/24`.
	<ip-addr>	An IPv4 address in a.b.c.d format.
host	A single source host.	
<host-address>	Single source host address.	
any	Matches any source IP address.	

Mode IPv4 Standard ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage An ACL can be configured with multiple ACL filters using sequence numbers. If the sequence number is omitted, the next available multiple of 10 will be used as the sequence number for the new filter. A new ACL filter can be inserted into the middle of an existing list by specifying the appropriate sequence number.

Note
The access control list being configured is selected by running the **access-list standard (named)** command with the required access control list number, or name, but with no further parameters selected.

Note
Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example To add a new entry accepting the IP network `10.1.1.0/24` at the sequence number `15` position, use the commands:

```
awplus# configure terminal
awplus(config)# access-list 99
awplus(config-ip-std-acl)# 15 permit 10.1.2.0 0.0.0.255
```

Related Commands
access-list (standard numbered)
show running-config
show ip access-list

clear ip prefix-list

Use this command to reset the hit count to zero in the prefix-list entries.

Syntax `clear ip prefix-list [<list-name>] [<ip-address>/<mask>]`

Parameter	Description
<list-name>	The name of the prefix-list.
<ip-address>/<mask>	The IP prefix and length.

Mode Privileged Exec

Example To clear a prefix-list named List1:

```
awplus# clear ip prefix-list List1
```

dos

Use this command to configure Denial-of-Service (DoS) features for a port. Six different DoS attacks can be detected: IP Options, Land, Ping-of-Death, Smurf, Synflood and Teardrop.

When the attack is detected, three different actions are available:

1. Shutdown the port for one minute
2. Cause an SNMP trap.
3. Send traffic to the mirror port

Syntax

```
dos {ipoptions|land|ping-of-death|smurf broadcast <ip-address>|
    synflood|teardrop} action {shutdown|trap|mirror}
```

Parameter	Description
`dos`	Denial-Of-Service.
`ipoptions`	IP Options attack.
`land`	Land attack.
`ping-of-death`	Large ping attack.
`smurf`	Ping to broadcast address.
`broadcast`	Broadcast.
<ip-address>	Local IP Broadcast Address.
`synflood`	SYN flood attack.
`teardrop`	IP fragmentation attack.
`action`	Action.
`shutdown`	Shutdown port.
`trap`	Trap to SNMP.
`mirror`	Send packets to mirror port.

Mode Interface Configuration for a switch port interface.

Default DoS attack detection is not configured by default on any switch port interface.

Usage See the below table for more information about the DoS attacks recognized by this command:

Type of DoS attack	Description
`ipoptions`	This type of attack occurs when an attacker sends packets containing bad IP options to a victim node. There are many different types of IP options attacks and this software does not try to distinguish between them. Rather, if this defense is activated, the number of ingress IP packets containing IP options is counted. If the number exceeds 20 packets per second, the switch considers this a possible IP options attack. This defense does not require the CPU to monitor packets, so does not put extra load on the switch's CPU.
`land`	This type of attack occurs when the Source IP and Destination IP address are the same. This can cause a target host to be confused. Since packets with the same source and destination addresses should never occur, these packets are dropped when this attack is enabled. This defense does not require the CPU to monitor packets, so does not put extra load on the switch's CPU.
`ping-of-death`	This type of attack results from a fragmented packet which, when reassembled, would exceed the maximum size of a valid IP datagram. To detect this attack, the final fragment of ICMP packets has to be sent to the CPU for inspection. This defense can therefore load the CPU. Note that the extra CPU load will not affect normal traffic switching between ports, but other protocols such as IGMP and STP may be affected. This defense is not recommended where a large number of fragmented packets are expected.
`smurf`	This type of attack is an ICMP ping packet to a broadcast address. Although routers should not forward packets to local broadcast addresses anymore (see RFC2644), the Smurf attack can still be explicitly discarded with this command. In order for the Smurf attack to work, the broadcast IP address is required. Any ICMP Ping packet with this destination address is considered an attack. This defense does not require the CPU to monitor packets, so does not put extra load on the switch's CPU.

Type of DoS attack	Description (cont.)
synflood	In this type of attack, an attacker, seeking to overwhelm a victim with TCP connection requests, sends a large number of TCP SYN packets with bogus source addresses to the victim. The victim responds with SYN ACK packets, but since the original source addresses are bogus, the victim node does not receive any replies. If the attacker sends enough requests in a short enough period, the victim may freeze operations once the requests exceed the capacity of its connections queue. To defend against this form of attack, a switch port monitors the number of ingress TCP-SYN packets it receives. An attack is recorded if a port receives more 60 TCP-SYN packets per second.
teardrop	In this DoS attack, an attacker sends a packet in several fragments with a bogus offset value, used to reconstruct the packet, in one of the fragments to a victim. This results in the victim being unable to reassemble the packet, possibly causing it to freeze operations.

Examples

To configure **smurf** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# dos smurf broadcast 192.168.1.0 action
shutdown
```

To configure **land** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# dos land action shutdown
```

To configure **ipoptions** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# dos ipoptions action shutdown
```

To configure **ping-of-death** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# dos ping-of-death action shutdown
```

To configure **synflood** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# dos synflood action shutdown
```

To configure **teardrop** DoS detection on `port1.0.1`, and shutdown the interface if an attack is detected, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# dos teardrop action shutdown
```

Related Commands **show dos interface**

ip prefix-list

Use this command to create an entry for an IPv4 prefix list.

Use the **no** variant of this command to delete the IPv4 prefix-list entry.

Syntax

```
ip prefix-list <list-name> [seq <1-429496725>] {deny|permit} {any|
   <ip-prefix>} [ge <0-32>] [le <0-32>]

ip prefix-list <list-name> description <text>

ip prefix-list sequence-number

no ip prefix-list <list-name> [seq <1-429496725>]

no ip prefix-list <list-name> [description <text>]

no ip prefix-list sequence-number
```

Parameter	Description
`<list-name>`	Specifies the name of a prefix list.
`seq <1-429496725>`	Sequence number of the prefix list entry.
`deny`	Specifies that the prefixes are excluded from the list.
`permit`	Specifies that the prefixes are included in the list.
`<ip-prefix>`	Specifies the IPv4 address and length of the network mask in dotted decimal in the format A.B.C.D/M.
`any`	Any prefix match. Same as **0.0.0.0/0 le 32**.
`ge <0-32>`	Specifies the minimum prefix length to be matched.
`le <0-32>`	Specifies the maximum prefix length to be matched.
`description <text>`	Text description of the prefix list.
`sequence-number`	Specify sequence numbers included or excluded in prefix list.

Mode Global Configuration

Usage When the device processes a prefix list, it starts to match prefixes from the top of the prefix list, and stops whenever a permit or deny occurs. To promote efficiency, use the **seq** parameter and place common permits or denials towards the top of the list. If you do not use the **seq** parameter, the sequence values are generated in a sequence of 5.

The parameters **ge** and **le** specify the range of the prefix lengths to be matched. When setting these parameters, set the **le** value to be less than 32, and the **ge** value to be less than or equal to the **le** value and greater than the ip-prefix mask length.

Prefix lists implicitly exclude prefixes that are not explicitly permitted in the prefix list. This means if a prefix that is being checked against the prefix list reaches the end of the prefix list without matching a permit or deny, this prefix will be denied.

Example To deny the IP addresses between `10.0.0.0/14` (`10.0.0.0 255.252.0.0`) and `10.0.0.0/22` (`10.0.0.0 255.255.252.0`) within the `10.0.0.0/8` (`10.0.0.0 255.0.0.0`) addressing range, enter the following commands:

```
awplus# configure terminal
awplus(config)# ip prefix-list mylist seq 12345 deny 10.0.0.0/
8 ge 14 le 22
```

Related Commands
match ip address
area filter-list
clear ip prefix-list
match route-type
show ip prefix-list

maximum-access-list

Sets the maximum number of filters that can be added to any access-list. These are access-lists within the ranges <1-199>, <1300-1999> and <2000-2699> and named standard and extended access-lists.

The **no** variant of this command removes the limit on the number of filters that can be added to a software access-list

Syntax

```
maximum-access-list <1-4294967294>
no maximum-access-list
```

Parameter	Description
<1-4294967294>	Filter range.

Mode Global Configuration

Example To set the maximum number of software filters to `200`:

```
awplus# configure terminal
awplus(config)# maximum-access-list 200
```

show access-list (IPv4 Software ACLs)

Use this command to display the specified access-list, or all access-lists if none have been specified. Note that only defined access-lists are displayed. An error message is displayed for an undefined access-list

Syntax

```
show access-list [<1-99>|<100-199>|<1300-1999>|<2000-2699>|
    <3000-3699>|<4000-4499>|<access-list-name>]
```

Parameter	Description
<1-99>	IP standard access-list.
<100-199>	IP extended access-list.
<1300-1999>	IP standard access-list (standard - expanded range).
<2000-2699>	IP extended access-list (extended - expanded range).
<3000-3699>	Hardware IP access-list.
<4000-4499>	Hardware MAC access-list.
<access-list-name>	IP named access-list.

Mode User Exec and Privileged Exec

Examples To show all access-lists configured on the switch:

```
awplus# show access-list
```

```
Standard IP access list 1
    deny 172.16.2.0, wildcard bits 0.0.0.255
Standard IP access list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
Hardware IP access list 3001
    permit ip 192.168.20.0 255.255.255.0 any
Hardware IP access list 3020
    permit tcp any 192.0.2.0/24
awplus#show access-list 20
```

To show the access-list with an ID of 20:

```
awplus# show access-list 20
```

```
Standard IP access-list 20
    deny 192.168.10.0, wildcard bits 0.0.0.255
    deny 192.168.12.0, wildcard bits 0.0.0.255
```

Note the below error message if you attempt to show an undefined access-list:

```
awplus# show access-list 2
```

```
% Can't find access-list 2
```

Related Commands

access-list standard (named)
access-list (standard numbered)
access-list (extended numbered)

show dos interface

Use this command to display the Denial-of-Service (DoS) features configured on a switch port interface from the **dos** command. See the **dos** command for descriptions of DoS attack types.

Syntax `show dos interface {<port-list>}`

Parameter	Description
<port-list>	Specify the switch port or port list to display DoS configuration options set with the **dos** command.

Mode Privileged Exec

Output **Figure 59-1: Example output from the show dos interface command prior to a DoS attack**

```
awplus#configure terminal
Enter configuration commands, one per line. End with CTNTL/Z.
awplus(config)#interface port1.0.1
awplus(config-if)#dos synflood action shutdown
awplus(config-if)#exit
awplus(config)#exit
awplus#show dos interface port1.0.1

DoS settings  for interface port1.0.1
----------------------------------------
Port status         : Enabled
ipoptions           : Disabled
land                : Disabled
ping-of-death       : Disabled
smurf               : Disabled
synflood            : Enabled
   Action           : Shutdown port
   Attacks detected : 0
teardrop            : Disabled
awplus#
```

Figure 59-2: Example output from the show dos interface command after a synflood DoS attack

```
awplus#show dos interface port1.0.1

DoS settings for interface port1.0.1
----------------------------------------
Port status         : Enabled
ipoptions           : Disabled
land                : Disabled
ping-of-death       : Disabled
smurf               : Disabled
synflood            : Enabled
   Action           : Shutdown port
   Attacks detected : 1
teardrop            : Disabled
awplus#
```

Table 59-5: Parameters in the show dos interface command output:

Type of DoS attack	Description
`Port status`	Displays `Enabled` when the port is configured as being administratively up after issuing the **no shutdown** command. Displays `Disabled` when the port is configured as being administratively down with the **shutdown** command.
`ipoptions`	Displays `Enabled` when the **ipoptions** parameter is configured with the **dos** command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **ipoptions** DoS attacks that have occurred on the interface. Displays `Disabled` when the **ipoptions** parameter is not configured with the **dos** command.
`land`	Displays `Enabled` when the **land** parameter is configured with the **dos** command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **land** DoS attacks that have occurred on the interface. Displays `Disabled` when the **land** parameter is not configured with the **dos** command.
`ping-of-death`	Displays `Enabled` when the **ping-of-death** parameter is configured with the **dos** command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **ping-of-death** DoS attacks that have occurred on the interface. Displays `Disabled` when the **ping-of-death** parameter is not configured with the **dos** command.
`smurf`	Displays `Enabled` when the **smurf** parameter is configured with the **dos** command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **smurf** DoS attacks that have occurred on the interface. Displays `Disabled` when the **smurf** parameter is not configured with the **dos** command.
`synflood`	Displays `Enabled` when the **synflood** parameter is configured with the **dos** command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **synflood** DoS attacks that have occurred on the interface. Displays `Disabled` when the **synflood** parameter is not configured with the **dos** command.
`teardrop`	Displays `Enabled` when the **teardrop** parameter is configured with the **dos** command, plus the action (`Shutdown port`, `Mirror port`, or `Trap port`) and the number of instances of any **teardrop** DoS attacks that have occurred on the interface. Displays `Disabled` when the **teardrop** parameter is not configured with the **dos** command.

Related Commands **dos**

show ip access-list

Use this command to display IP access-lists.

Syntax

```
show ip access-list [<1-99>|<100-199>|<1300-1999>|<2000-2699>|
    <access-list-name>]
```

Parameter	Description
<1-99>	IP standard access-list.
<100-199>	IP extended access-list.
<1300-1999>	IP standard access-list (expanded range).
<2000-2699>	IP extended access-list (expanded range).
<access-list-name>	IP named access-list.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip access-list
```

Output **Figure 59-3: Example output from the show ip access-list command**

```
Standard IP access-list 1
    permit 172.168.6.0, wildcard bits 0.0.0.255
    permit 192.168.6.0, wildcard bits 0.0.0.255
```

show ip prefix-list

Use this command to display the IPv4 prefix-list entries. Note that this command is valid for RIP and BGP routing protocols only.

Syntax `show ip prefix-list [<name>|detail|summary]`

Parameter	Description
<name>	Specify the name of a prefix list in this placeholder.
`detail`	Specify this parameter to show detailed output for all IPv4 prefix lists.
`summary`	Specify this parameter to show summary output for all IPv4 prefix lists.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip prefix-list
awplus# show ip prefix-list 10.10.0.98/8
awplus# show ip prefix-list detail
```

Related Commands ip prefix-list

Chapter 60: IPv6 Hardware Access Control List (ACL) Commands

Introduction 60.2
IPv6 Hardware Access List Commands and Prompts 60.3
Command List 60.4
commit (IPv6) 60.4
ipv6 access-list (named) 60.5
(ipv6 access-list named ICMP filter) 60.7
(ipv6 access-list named protocol filter) 60.10
(ipv6 access-list named TCP UDP filter) 60.14
ipv6 traffic-filter 60.17
show ipv6 access-list (IPv6 Hardware ACLs) 60.18

Introduction

This chapter provides an alphabetical reference for the IPv6 Hardware Access Control List (ACL) commands, and contains detailed command information and command examples about IPv6 hardware ACLs, which are applied directly to interfaces using the **ipv6 traffic-filter** command.

Note

See **Chapter 57, Access Control Lists Introduction** for descriptions of ACLs, and for further information about rules when applying ACLs see the **ACL Rules** section.

See **ACL Filter Sequence Numbers** and **ACL Filter Sequence Number Behavior** sections in **Chapter 57, Access Control Lists Introduction** about ACL Filters.

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself. For more information on link aggregation see **Chapter 22, Link Aggregation Introduction and Configuration**, and **Chapter 23, Link Aggregation Commands**.

Note that text in parenthesis in command names indicates usage not keyword entry. For example, **ipv6-access-list (named)** indicates named IPv6 ACLs entered as `ipv6-access-list <name>` where `<name>` is a placeholder not a keyword.

Note also that parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI, such as **(ipv6 access-list standard IPv6 filter)** represents command entry in the format shown in the syntax `[<sequence-number>] {deny|permit} {<IPv6-source-address/prefix-length>|any}`.

Note

Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

IPv6 Hardware Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table **"IPv6 Hardware Access List Commands and Prompts"** shows the CLI prompts at which ACL commands are entered.

Table 60-1: IPv6 Hardware Access List Commands and Prompts

Command Name	Command Mode	Prompt
show ipv6 access-list (IPv6 Hardware ACLs)	Privileged Exec	`awplus#`
ipv6 access-list (named)	Global Configuration	`awplus(config)#`
(ipv6 access-list named ICMP filter)	Global Configuration	`awplus(config)#`
ipv6 traffic-filter	Interface Configuration	`awplus(config-if)#`
commit (IPv6)	IPv6 Hardware ACL Configuration	`awplus(config-ipv6-hw-acl)#`
(ipv6 access-list named ICMP filter)	IPv6 Hardware ACL Configuration	`awplus(config-ipv6-hw-acl)#`
(ipv6 access-list named protocol filter)	IPv6 Hardware ACL Configuration	`awplus(config-ipv6-hw-acl)#`
(ipv6 access-list named TCP UDP filter)	IPv6 Hardware ACL Configuration	`awplus(config-ipv6-hw-acl)#`

Command List

commit (IPv6)

Use this command to commit the IPv6 ACL filter configuration entered at the console to the hardware immediately without exiting the IPv6 Hardware ACL Configuration mode.

This command forces the associated hardware and software IPv6 ACLs to synchronize.

Syntax `commit`

Mode IPv6 Hardware ACL Configuration

Usage Normally, when an IPv6 hardware ACL is edited, the new configuration state of the IPv6 ACL is not written to hardware until you exit IPv6 Hardware ACL Configuration mode. By entering this command you can ensure that the current state of a hardware access-list that is being edited is written to hardware immediately.

Scripts typically do not include the **exit** command to exit configuration modes, potentially leading to IPv6 ACL filters in hardware not being correctly updated. Using this **commit** command in a configuration script after specifying an IPv6 hardware ACL filter ensures that it is updated in the hardware.

Example To update the hardware with the IPv6 ACL filter configuration, use the command:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-ipv6-acl
awplus(config-ipv6-hw-acl)# commit
```

Related Commands **ipv6 access-list (named)**

ipv6 access-list (named)

Use this command to either create a new IPv6 hardware access-list, or to select an existing IPv6 hardware access-list in order to apply a filter entry to it.

Use the **no** variant of this command to delete an existing IPv6 hardware access-list.

Note Before you can delete an access-list, you must first remove it from any interface it is assigned to.

Syntax

```
ipv6 access-list <ipv6-access-list-name>
no ipv6 access-list <ipv6-access-list-name>
```

Parameter	Description
<ipv6-access-list-name>	Specify an IPv6 access-list name.

Mode Global Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicitly match a filter is permitted.

Usage Use IPv6 hardware named access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 hardware named access-list when a match is encountered.

For backwards compatibility you can either create IPv6 hardware named access-lists from within this command, or you can enter `ipv6 access-list` followed by only the IPv6 hardware named access-list name. This latter (and preferred) method moves you to the `(config-ipv6-hw-acl)` prompt for the selected IPv6 hardware named access-list number, and from here you can configure the filters for this selected IPv6 hardware named access-list.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To create an IPv6 access-list named `my-ipv6-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-ipv6-acl
awplus(config-ipv6-hw-acl)#
```

To delete the IPv6 access-list named `my-ipv6-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# no ipv6 access-list my-ipv6-acl
```

Validation Commands **show ipv6 access-list (IPv6 Hardware ACLs)**

Related Commands **(ipv6 access-list named ICMP filter)**
(ipv6 access-list named protocol filter)
(ipv6 access-list named TCP UDP filter)
ipv6 traffic-filter
show ipv6 access-list (IPv6 Hardware ACLs)

(ipv6 access-list named ICMP filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with ICMP (Internet Control Message Protocol) packets, to the current named IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

Note that specifying the **send-to-cpu** parameter could result in EPSR healthcheck messages and other control packets being dropped.

As an optional parameter **vlan** can be matched for tagged (802.1q) packet.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with ICMP (Internet Control Message Protocol) packets, from the current named IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Syntax [ip|icmp]

```
[<sequence-number> {deny|permit|send-to-cpu|send-to-mirror|
   copy-to-cpu|copy-to-mirror} {ipv6|icmp} {<ipv6-source-address/
   prefix-length>|<ipv6-source-address> <ipv6-source-wildcard>|
   host <ipv6-source-host>|any} {<ipv6-destination-address/
   prefix-length>|<ipv6-destintation-addr>
   <ipv6-destination-wildcard>|host <ipv6-destination-host-address>|
   any} [<icmp-type>][vlan <1-4094>]

no [<sequence-number>]{deny|permit|send-to-cpu|send-to-mirror|
   copy-to-cpu|copy-to-mirror} {ip|icmp}{<ipv6-source-address/
   prefix-length>|<ipv6-source-address> <ipv6-source-wildcard>|
   host <ipv6-source-host>|any} {<ipv6-destination-address/
   prefix-length>|<ipv6-destination-address>
   <ipv6-destination-wildcard>|host <ipv6-destination-host-address>|
   any} [<icmp-type>][vlan <1-4094>]

no [<sequence-number>]
```

Parameter	Description
<sequence-number>	<1-65535> *The sequence number for the filter entry of the selected access control list.*
deny	Specifies the packets to reject.
permit	Specifies the packets to permit.
send-to-cpu	Specifies the packets to send to the CPU. Specifying this parameter could result in EPSR healthcheck messages and other control packets being dropped.
send-to-mirror	Specifies the packets to send to mirror port.
copy-to-cpu	Specifies the packets to copy to the CPU.
copy-to-mirror	Specifies the packets to copy to the mirror port.

Parameter(cont.)	Description(cont.)
ipv6	IPv6 packet, defined by the following parameters.
icmp	ICMP packet, defined by the following parameters.
<ipv6-source-address/ prefix-length>	Specifies a source address and prefix length. The IPv6 address prefix uses the format X:X::/ prefix-length. The prefix-length is usually set between 0 and 64.
<ipv6-source-address>	Specifies the IPv6 source address. The IPv6 address uses the format X:X::X:X.
<ipv6-source-wildcard>	Specifies source wildcard bits in IPv6 format X:X::X:X. Note that binary 1 represents a don't care condition, and binary 0 represents a match.
host *<ipv6-source-host>*	Specifies a single source host address. The IPv6 address uses the format X:X::X:X.
any	Specifies any Source host.
<ipv6-destination-address/prefix-length>	Specifies a destination address and prefix length. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
<ipv6-destination-addrress>	Specifies a destination address. The IPv6 address uses the format X:X::X:X.
<ipv6-destination-wildcard>	Specify destination wildcard bits in IPv6 format X:X::X:X.
host *<ipv6-destination-host-address>*	Specify a single destination host address. The IPv6 address uses the format X:X::X:X.
any	Specifies any destination host.
<icmp-type>	Optional. Specify to filter by ICMP message type number. Valid values are from 0 to 255.
vlan	This parameter can be used in either single or double-tagged VLAN networks. It is the conventional VLAN tag (VID). In a double-tagged network it is sometimes referred to as the STAG.
<1-4094>	The VLAN VID.

Mode IPv6 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicate match a filter is permitted.

Examples

To specify a hardware IPv6 access-list named `my-acl1` and add an ACL filter entry that blocks all ICMP6 echo requests, enter the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl1
awplus(config-ipv6-hw-acl)# deny icmp any any icmp-type 128
```

To specify a hardware IPv6 access-list named `my-acl2` and add an ACL filter entry that blocks all ICMP6 echo requests on the default VLAN (`vlan1`), enter the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl2
awplus(config-ipv6-hw-acl)# deny icmp any any icmp-type 128
vlan 1
```

To remove an ACL filter entry that blocks all ICMP6 echo requests from he hardware IPv6 access-list named `my-acl1`, enter the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl1
awplus(config-ipv6-hw-acl)# no deny icmp any any icmp-type 128
```

Validation Commands

show ipv6 access-list (IPv6 Hardware ACLs)

Related Commands

ipv6 access-list (named)
(ipv6 access-list named protocol filter)
(ipv6 access-list named TCP UDP filter)
ipv6 traffic-filter
show ipv6 access-list (IPv6 Hardware ACLs)

(ipv6 access-list named protocol filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with an IP protocol type specified, to the current named IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

Note that specifying the **send-to-cpu** parameter could result in EPSR healthcheck messages and other control packets being dropped.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with an IP protocol type specified, from the current named IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Syntax

```
[<sequence-number> {deny|permit|send-to-cpu|send-to-mirror|
   copy-to-cpu|copy-to-mirror} {ipv6|proto <ip-protocol>}
   {<ipv6-source-prefix/prefix-length>|<ipv6-source-address>
   <ipv6-source-wildcard>|host <ipv6-source-host>|any}
   {<ipv6-destination-prefix/prefix-length>|
   <ipv6-destination-address> <ipv6-destination-wildcard>|
   host <ipv6-destination-host>|any}

[<sequence-number>]no {deny|permit|send-to-cpu|send-to-mirror|
   copy-to-cpu|copy-to-mirror} {ipv6|proto <ip-protocol>}
   {<ipv6-source-prefix/prefix-length>|<ipv6-source-address>
   <ipv6-source-wildcard>|host <ipv6-source-host>|any}
   {<ipv6-destination-prefix/prefix-length>|
   <ipv6-destination-address> <ipv6-destination-wildcard>|
   host <ipv6-destination-host>|any}

no [<sequence-number>]
```

Parameter	Description
<sequence-number>	<1-65535> *The sequence number for the filter entry of the selected access control list.*
deny	Specifies packets to reject.
permit	Specifies packets to permit.
send-to-cpu	Specifies packets to send to the CPU. Specifying this parameter could result in EPSR healthcheck messages and other control packets being dropped.
send-to-mirror	Specifies packets to send to mirror port.
copy-to-cpu	Specifies packets to copy to the CPU.
copy-to-mirror	Specifies packets to copy to the mirror port.
ipv6	Specifies IPv6 packet.
proto *<ip-protocol>*	Specify IP protocol number <1-255>.

Parameter (cont.)	Description (cont.)
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**)

Protocol Number	Protocol Description [RFC Reference]
1	Internet Control Message [RFC792]
2	Internet Group Management [RFC1112]
3	Gateway-to-Gateway [RFC823]
4	IP in IP [RFC2003]
5	Stream [RFC1190] [RFC1819]
6	TCP (Transmission Control Protocol) [RFC793]
8	EGP (Exterior Gateway Protocol) [RFC888]
9	IGP (Interior Gateway Protocol) [IANA]
11	Network Voice Protocol [RFC741]
17	UDP (User Datagram Protocol) [RFC768]
20	Host monitoring [RFC869]
27	RDP (Reliable Data Protocol) [RFC908]
28	IRTP (Internet Reliable Transaction Protocol) [RFC938]
29	ISO-TP4 (ISO Transport Protocol Class 4) [RFC905]
30	Bulk Data Transfer Protocol [RFC969]
33	DCCP (Datagram Congestion Control Protocol) [RFC4340]
48	DSR (Dynamic Source Routing Protocol) [RFC4728]
50	ESP (Encap Security Payload) [RFC2406]
51	AH (Authentication Header) [RFC2402]
54	NARP (NBMA Address Resolution Protocol) [RFC1735]
58	ICMP for IPv6 [RFC1883]
59	No Next Header for IPv6 [RFC1883]
60	Destination Options for IPv6 [RFC1883]
88	EIGRP (Enhanced Interior Gateway Routing Protocol)
89	OSPFIGP [RFC1583]

Parameter (cont.)	Description (cont.)	
<ip-protocol> (cont.)	97	Ethernet-within-IP Encapsulation / RFC3378
	98	Encapsulation Header / RFC1241
	108	IP Payload Compression Protocol / RFC2393
	112	Virtual Router Redundancy Protocol / RFC3768
	134	RSVP-E2E-IGNORE / RFC3175
	135	Mobility Header / RFC3775
	136	UDPLite / RFC3828
	137	MPLS-in-IP / RFC4023
	138	MANET Protocols / RFC-ietf-manet-iana-07.txt
	139-252	Unassigned / IANA
	253	Use for experimentation and testing / RFC3692
	254	Use for experimentation and testing / RFC3692
	255	Reserved / IANA
<ipv6-source-prefix/prefix-length>	Specify source address and mask. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.	
<ipv6-source-address>	Specifies the source address. The IPv6 address uses the format X:X::X:X.	
<ipv6-source-wildcard>	Specifies the source wildcard bits, in IPv6 format X:X::X:X .	
host *<ipv6-source-host>*	Specifies a single source host. The IPv6 address uses the format X:X::X:X.	
any	Specifies any source host. An abbreviation for the IPv6 prefix *::/0*	
<ipv6-dest-prefix/prefix-length>	Specifies a destination address and mask. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64.	
<ipv6-destination-address>	Specify destination address. The IPv6 address uses the format X:X::X:X.	
<ipv6-destination-wildcard>	Specify destination wildcard bits in IPv6 format X:X::X:X	
host *<ipv6-destination-host>*	Specify a single destination host address. The IPv6 address uses the format X:X::X:X.	
any	Specifies any destination host. An abbreviation for the IPv6 prefix *::/0*	

Parameter (cont.)	Description (cont.)
`vlan`	This parameter can be used in either single or double-tagged VLAN networks. It is the conventional VLAN tag (VID). In a double-tagged network it is sometimes referred to as the STAG.
<1-4094>	The VLAN VID.

Mode IPv6 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicate match a filter is permitted.

Usage This command adds a hardware classification filter (for use with features such as QoS), to a current standard IPv6 access-list. The filter will match on any IP protocol type packet that has the specified source and destination IPv6 addresses and the specified IP protocol type. The parameter `any` may be specified if an address does not matter,

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To add an ACL filter entry to block IP traffic from network `2001:0db8::0/64` to the hardware IPv6 access-list named `my-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl
awplus(config-ipv6-hw-acl)# deny ipv6 2001:0db8::0/64
```

To remove an ACL filter entry that blocks all IPv6 traffic from network `2001:0db8::0/64` from the hardware IPv6 access-list named `my-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl
awplus(config-ipv6-hw-acl)# no deny ipv6 2001:0db8::0/64
```

Validation Commands **show ipv6 access-list (IPv6 Hardware ACLs)**

Related Commands **ipv6 access-list (named)**
(ipv6 access-list named ICMP filter)
(ipv6 access-list named TCP UDP filter)
ipv6 traffic-filter
show ipv6 access-list (IPv6 Hardware ACLs)

(ipv6 access-list named TCP UDP filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) source and destination ports specified, to the current named IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

Note that specifying the **send-to-cpu** parameter could result in EPSR healthcheck messages and other control packets being dropped.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with TCP or UDP source and destination ports specified, from the current named IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Syntax

```
[<sequence-number>]{deny|permit|send-to-cpu|send-to-mirror|
   copy-to-cpu|copy-to-mirror} {tcp|udp} {<ipv6-source-prefix/
   prefix-length>|<ipv6-source-address> <ipv6-source-wildcard>|
   host <ipv6-source-host>|any} {eq <sourceport>|lt <sourceport>|
   gt <sourceport>|ne <sourceport>}|[range <start-range>
   <end-range>] {<ipv6-destination-prefix/prefix-length>|
   <ipv6-destination-address> <ipv6-destination-wildcard>|
   host <ipv6-destination-host>|any} {[eq <destport>|lt <destport>|
   gt <destport>|ne <destport>}[range <start-range> <end-range>]

no {deny|permit|send-to-cpu|send-to-mirror|copy-to-cpu|
   copy-to-mirror} {tcp|udp} {<ipv6-source-prefix/prefix-length>|
   <ipv6-source-address> <ipv6-source-wildcard>|
   host <ipv6-source-host>|any} {eq <sourceport>|lt <sourceport>|
   gt <sourceport>|ne <sourceport>}|[range <start-range>
   <end-range>] {<ipv6-destination-prefix/prefix-length>|
   <ipv6-destination-address> <ipv6-destination-wildcard>|
   host <ipv6-destination-host>|any} {eq <destport>|lt <destport>|
   gt <destport>|ne <destport>}|[range <start-range> <end-range>]

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> *The sequence number for the filter entry of the selected access control list.*
deny	Specify packets to reject.
permit	Specifies the packets to permit.
send-to-cpu	Specifies the packets to send to the CPU. Specifying this parameter could result in EPSR healthcheck messages and other control packets being dropped.
send-to-mirror	Specifies the packets to send to mirror port.
copy-to-cpu	Specifies the packets to copy to the CPU.
copy-to-mirror	Specifies the packets to copy to the mirror port.
tcp	Specifies a TCP packet.

Parameter (cont.)	Description (cont.)
udp	Specifies a UDP packet.
<ipv6-source-prefix/ prefix-length>	Specifies the source address with mask. The IPv6 address prefix uses the format X:X::/ prefix-length. The prefix-length is usually set between 0 and 64.
<ipv6-source-address>	Specifies the source address. The IPv6 address uses the format X:X::X:X.
<ipv6-source-wildcard>	Specifies the source wildcard bits in IPv6 format X:X::X:X.
host *<ipv6-source-host>*	Specifies the a single source host. The IPv6 address uses the format X:X::X:X.
any	Specifies any Source host. An abbreviation for the IPv6 prefix *::/0*.
eq	Equal to.
lt	Less than.
gt	Greater than.
ne	Not equal to.
<sourceport>	The source port number, specified as an integer between 0 and 65535.
<destport>	The destination port number, specified as an integer between 0 and 65535.
range	Range of port numbers. Match only packets within range.
<start-range>	The port number at the start of the range *<0-65535>*.
<end-range>	The port number at the end of the range *<0-65535>*.
<ipv6-dest-prefix/ prefix-length>	Specify destination address with mask. The IPv6 address prefix uses the format X:X::/ prefix-length. The prefix-length is usually set between 0 and 64.
<ipv6-destination-address>	Specify destination address. The IPv6 address uses the format X:X::X:X.
<ipv6-destination-wildcard>	Specify destination wildcard bits in IPv6 format X:X::X:X .
host *<ipv6-destination-host>*	Specify a single destination host address. The IPv6 address uses the format X:X::X:X.
any	Specifies any destination host. An abbreviation for the IPv6 prefix **::/0**.

Mode IPv6 Hardware ACL Configuration

Default Any traffic on an interface controlled by a hardware ACL that does not explicate match a filter is permitted.

Usage The filter entry will match on any TCP or UDP packet that has the specified source and destination IPv6 addresses and TCP or UDP type. The parameter `any` may be specified if an address does not matter.

Note Hardware ACLs will **permit** access unless **explicitly denied** by an ACL action.

Examples To add an ACL filter entry that blocks all SSH traffic from network `2001:0db8::0/64` to the hardware IPv6 access-list named `my-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl
awplus(config-ipv6-hw-acl)# deny tcp 2001:0db8::0/64 any eq 22
```

To add an ACL filter entry that blocks all SSH traffic from network `2001:0db8::0/64` on the default VLAN (`vlan1`) to the hardware IPv6 access-list named `my-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl
awplus(config-ipv6-hw-acl)# deny tcp 2001:0db8::0/64 any eq 22
vlan 1
```

To remove an ACL filter entry that blocks all SSH traffic from network `2001:0db8::0/64` from the hardware IPv6 access-list named `my-acl`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list my-acl
awplus(config-ipv6-hw-acl)# no deny tcp 2001:0db8::0/64 any eq
22
```

Validation Commands **show ipv6 access-list (IPv6 Hardware ACLs)**

Related Commands **ipv6 access-list (named)**
(ipv6 access-list named ICMP filter)
(ipv6 access-list named protocol filter)
ipv6 traffic-filter
show ipv6 access-list (IPv6 Hardware ACLs)

ipv6 traffic-filter

This command adds an IPv6 hardware-based access-list to an interface. The number of access-lists that can be added is determined by the amount of available space in the hardware-based packet classification tables.

Use the **no** variant of this command to remove an IPv6 hardware-based access-list from an interface.

Syntax

```
ipv6 traffic-filter <ipv6-access-list-name>
no ipv6 traffic-filter <ipv6-access-list-name>
```

Parameter	Description
<ipv6-access-list-name>	Hardware IPv6 access-list name.

Mode Interface Configuration (to apply an IPv6 hardware ACL to a specific switch port).

Usage This command adds an IPv6 hardware-based access-list to an interface. The number of access-lists that can be added is determined by the amount of available space in the hardware-based packet classification tables.

To apply the access-list to all ports on the switch, execute the command in the Global Configuration mode. To apply the access-list to a Layer 2 interface or Layer 2 interface range, apply the command in the Interface Configuration mode. See the examples for each mode below.

Examples To add access-list `acl1` as a traffic-filter to interface `port1.0.1`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# ipv6 traffic-filter acl1
```

To remove access-list `acl1` as a traffic-filter from interface `port1.0.1`, enter the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no ipv6 traffic-filter acl1
```

Related Commands

ipv6 access-list (named)
(ipv6 access-list named ICMP filter)
(ipv6 access-list named protocol filter)
(ipv6 access-list named TCP UDP filter)
ipv6 traffic-filter
show ipv6 access-list (IPv6 Hardware ACLs)

show ipv6 access-list (IPv6 Hardware ACLs)

Use this command to display all configured hardware IPv6 access-lists or the IPv6 access-list specified by name. Omitting the optional access-list name parameter will display all IPv6 ACLs.

Use the **show ipv6 access-list standard** command to display the IPv6 access-list specified by name as defined from the **ipv6 access-list (named)** command.

Syntax

```
show ipv6 access-list [<access-list-name>]
show ipv6 access-list standard [<access-list-name>]
```

Parameter	Description
`standard`	Named standard access-list.
`<access-list-name>`	Hardware IPv6 access-list name.

Mode User Exec and Privileged Exec

Examples To show the standard named ipv6 access-list acl_name use the following command:

```
awplus# show ipv6 access-list standard acl_name
```

Output **Figure 60-1: Example output from the show ipv6 access-list standard command**

```
Named Standard IPv6 access-list acl_name
  deny any
```

To show all configured ipv6 access-lists use the command:

```
awplus# show ipv6 access-list
```

Output **Figure 60-2: Example output from the show ipv6 access-list command**

```
IPv6 access-list deny_icmp
  deny icmp any any vlan 1

IPv6 access-list deny_ssh
  deny tcp abcd::0/64 any eq 22
```

Related Commands
ipv6 access-list (named)
(ipv6 access-list named ICMP filter)
(ipv6 access-list named protocol filter)
(ipv6 access-list named TCP UDP filter)
ipv6 traffic-filter

Chapter 61: IPv6 Software Access Control List (ACL) Commands

Introduction 61.2
IPv6 Software Access List Commands and Prompts 61.3
Command List 61.4
ipv6 access-list extended (named) 61.4
ipv6 access-list extended proto 61.8
(ipv6 access-list extended IP protocol filter) 61.11
(ipv6 access-list extended TCP UDP filter) 61.14
ipv6 access-list standard (named) 61.16
(ipv6 access-list standard filter) 61.18
ipv6 prefix-list 61.20
show ipv6 access-list (IPv6 Software ACLs) 61.22
show ipv6 prefix-list 61.23

Introduction

This chapter provides an alphabetical reference for the IPv6 Software Access Control List (ACL) commands, and contains detailed command information and command examples about IPv6 software ACLs as applied to Routing and Multicasting, which are not applied to interfaces.

Note

See **Chapter 57, Access Control Lists Introduction** for descriptions of ACLs, and for further information about rules when applying ACLs see the **ACL Rules** section.

See **ACL Filter Sequence Numbers** and **ACL Filter Sequence Number Behavior** sections in **Chapter 57, Access Control Lists Introduction** about ACL Filters.

See all relevant Routing commands and configurations in **"Layer Three Switching and Routing"** and all relevant Multicast commands and configurations in **"Multicast Applications"**.

To apply ACLs to an LACP channel group, apply it to all the individual switch ports in the channel group. To apply ACLs to a static channel group, apply it to the static channel group itself. For more information on link aggregation see **Chapter 22, Link Aggregation Introduction and Configuration**, and **Chapter 23, Link Aggregation Commands**.

Note that text in parenthesis in command names indicates usage not keyword entry. For example, **ipv6-access-list (named)** indicates named IPv6 ACLs entered as `ipv6-access-list <name>` where `<name>` is a placeholder not a keyword.

Note also that parenthesis surrounding ACL filters indicates the type of ACL filter not the keyword entry in the CLI. For example, **(ipv6 access-list standard IPv6 filter)** represents command entry in the format:

`[<sequence-number>] {deny|permit} {<IPv6-source-address/prefix-length>|any}.`

Note

Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

IPv6 Software Access List Commands and Prompts

Many of the ACL commands operate from sub-modes that are specific to particular ACL types. The table **"IPv6 Software Access List Commands and Prompts"** shows the CLI prompts at which ACL commands are entered.

Table 61-1: IPv6 Software Access List Commands and Prompts

Command Name	Command Mode	Prompt
show ipv6 access-list (IPv6 Software ACLs)	Privileged Exec	`awplus#`
ipv6 access-list extended (named)	Global Configuration	`awplus(config)#`
ipv6 access-list standard (named)	Global Configuration	`awplus(config)#`
(ipv6 access-list extended IP protocol filter)	IPv6 Extended ACL Configuration	`awplus(config-ipv6-ext-acl)#`
(ipv6 access-list extended TCP UDP filter)	IPv6 Extended ACL Configuration	`awplus(config-ipv6-ext-acl)#`
(ipv6 access-list standard filter)	IPv6 Standard ACL Configuration	`awplus(config-ipv6-std-acl)#`

Command List

ipv6 access-list extended (named)

Use this command when configuring an IPv6 extended access-list for filtering frames that permit or deny IP, ICMP, TCP, UDP packets or ICMP packets with a specific value based on the source or destination.

The **no** variant of this command removes a specified IPv6 extended access-list.

Syntax [list-name]

```
ipv6 access-list extended <list-name>

no ipv6 access-list extended <list-name>
```

Parameter	Description
<list-name>	A user-defined name for the IPv6 software extended access-list.

Syntax [any|icmp|ip]

```
ipv6 access-list extended <list-name> {deny|permit} {any|icmp|ip}
   {<ipv6-source-address/prefix-length>|any}
   {<ipv6-destination-address/prefix-length>|any}
   [<icmp-type <icmp-type>][log]

no ipv6 access-list extended <list-name> {deny|permit} {any|icmp|ip}
   {<ipv6-source-address/prefix-length>|any}
   {<ipv6-destination-address/prefix-length>|any}
   [<icmp-type <icmp-type>][log]
```

Syntax [tcp|udp]

```
ipv6 access-list extended <list-name> {deny|permit} {tcp|udp}
   {<ipv6-source-address/prefix-length>|any} {eq <sourceport>|
   lt <sourceport>|gt <sourceport>|
   ne <sourceport>}{<ipv6-destination-address/prefix-length>|any}
   {eq <destport>|lt <destport>|gt <destport>|ne <destport>} [log]

no ipv6 access-list extended <list-name> {deny|permit} {tcp|udp}
   {<ipv6-source-address/prefix-length>|any} {eq <sourceport>|
   lt <sourceport>|gt <sourceport>|
   ne <sourceport>}{<ipv6-destination-addr/prefix-length>|any}
   {eq <destport>|lt <destport>|gt <destport>|ne <destport>} [log]
```

Parameter	Description
<list-name>	A user-defined name for the IPv6 software extended access-list.
deny	The IPv6 software extended access-list rejects packets that match the type, source, and destination filtering specified with this command.
permit	The IPv6 software extended access-list permits packets that match the type, source, and destination filtering specified with this command.
any	For ICMP|IP The IPv6 software extended access-list matches any type of packet.

Parameter (cont.)	Description (cont.)
`ip`	For ICMP|IP The IPv6 software extended access-list matches only IP packets.
`icmp`	For ICMP|IP The IPv6 software extended access-list matches only ICMP packets.
`tcp`	For TCP/UDP The IPv6 software extended access-list matches only TCP packets.
`udp`	For TCP/UDP The IPv6 software extended access-list matches only UDP packets.
<ipv6-source-address/prefix-length>	Specifies a source address and prefix length. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64.
<ipv6-destination-address/prefix-length>	Specifies a destination address and prefix length. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
`any`	Matches any IPv6 address.
<sourceport>	For TCP/UDP The source port number, specified as an integer between 0 and 65535.
<destport>	For TCP/UDP The destination port number, specified as an integer between 0 and 65535.
`icmp-type`	For ICMP|IP Matches only a specified type of ICMP messages. This is valid only when the filtering is set to match ICMP packets.
`eq`	For TCP/UDP Matches port numbers equal to the port number specified immediately after this parameter.
`lt`	For TCP/UDP Matches port numbers less than the port number specified immediately after this parameter.
`gt`	For TCP/UDP Matches port numbers greater than the port number specified immediately after this parameter.

Parameter (cont.)	Description (cont.)	
`ne`	For TCP/UDP Matches port numbers not equal to the port number specified immediately after this parameter.	
<icmp-type>	For ICMP|IP The ICMP type, as defined in RFC792 and RFC950. Specify one of the following integers to create a filter for the ICMP message type:	
	0	Echo replies.
	3	Destination unreachable messages.
	4	Source quench messages.
	5	Redirect (change route) messages.
	8	Echo requests.
	11	Time exceeded messages.
	12	Parameter problem messages.
	13	Timestamp requests.
	14	Timestamp replies.
	15	Information requests.
	16	Information replies.
	17	Address mask requests.
	18	Address mask replies.
`log`	Logs the results.	

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use IPv6 extended access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 extended access-list when a match is encountered.

For backwards compatibility you can either create IPv6 extended access-lists from within this command, or you can enter `ipv6 access-list extended` followed by only the IPv6 extended access-list name. This latter (and preferred) method moves you to the `(config-ipv6-ext-acl)` prompt for the selected IPv6 extended access-list number, and from here you can configure the filters for this selected access-list.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example 1 [creating a list]

To add a new filter to the access-list named `my-list` that will reject incoming ICMP packets from `2001:0db8::0/64` to `2001:0db8::f/64`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# icmp 2001:0db8::0/64 2001:0db8::f/64
```

Example 2 [adding to a list]

To insert a new filter at sequence number `5` of the access-list named `my-list` that will accept ICMP type 8 packets from the `2001:0db8::0/64` network to the `2001:0db8::f/64` network, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 icmp 2001:0db8::0/64
2001:0db8::f/64
```

Example 3 [list with filter]

To create the access-list named `TK` to deny TCP protocols, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended TK deny tcp any eq
14 any lt 12 log
```

Related Commands

ipv6 access-list extended proto (ipv6 access-list extended IP protocol filter)
ipv6 access-list extended TCP UDP filter)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

ipv6 access-list extended proto

Use this command when configuring an IPv6 extended access-list for filtering frames that permit or deny packets with a specific value based on the IP protocol number specified.

The **no** variant of this command removes a specified IPv6 extended access-list with an IP protocol number.

Syntax

```
ipv6 access-list extended <list-name> {deny|permit}
   proto <ip-protocol>} {<ipv6-source-address/prefix>|any}
   {<ipv6-destination-address/prefix>|any} [log]

no ipv6 access-list extended <list-name> {deny|permit}
   proto <ip-protocol>{<ipv6-source-address/prefix>|any}
   {<ipv6-destination-address/prefix>|any} [log]
```

Parameter	Description
<list-name>	A user-defined name for the IPv6 software extended access-list.
deny	Specifies the packets to reject.
permit	Specifies the packets to accept.
proto	The IP Protocol type specified by it protocol number <1-255>.
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**).
	Protocol Number
	1
	2
	3
	4
	5
	6
	8

Parameter (cont.)	Description (cont.)
<ip-protocol> (cont.)	**Protocol Number**
	9
	11
	17
	20
	27
	28
	29
	30
	33
	48
	50
	51
	54
	58
	59
	60
	88
	89
	97
	98
	108
	112
	134
	135
	136
	137
	138
	139-252
	253
	254
	255
<ipv6-source-address/prefix>	IPv6 source address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
any	Any source address or local address.

Parameter (cont.)	Description (cont.)
<ipv6-destination-address/prefix>	IPv6 destination address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
`any`	Any destination address or remote address.
`log`	Log the results.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use IPv6 extended access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 extended access-list when a match is encountered.

The filter entry will match on any IP protocol type packet that has the specified source and destination IPv6 addresses and the specified IP protocol type. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To create the IPv6 access-list named `ACL-1` to deny IP protocol `9` packets from `2001:0db8:1::1/128` to `2001:0db8:f::1/128`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended ACL-1 deny proto 9
                2001:0db8:1::1/128 2001:0db8:f::1/128
```

To remove the IPv6 access-list named `ACL-1` to deny IP protocol `9` packets from `2001:0db8:1::1/128` to `2001:0db8:f::1/128`, use the commands:

```
awplus# configure terminal
awplus(config)# no ipv6 access-list extended ACL-1 deny proto
                10 2001:0db8:1::1/128 2001:0db8:f::1/128
```

Related Commands
ipv6 access-list extended (named)
(ipv6 access-list extended IP protocol filter)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

(ipv6 access-list extended IP protocol filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with or without an IP protocol specified, to the current extended IPv6 access-list. If a sequence is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with or without an IP protocol filter entry, from the current extended IPv6 access-list. You can specify the ACL filter entry by entering either its sequence number, or its filter entry profile.

Syntax [ip|proto]

```
[<sequence-number>] {deny|permit} {ip|any|proto <ip-protocol>}
    {<ipv6-source-address/prefix>|any} {<ipv6-destination-address/
    prefix>|any} [log]

no {deny|permit} {ip|any|proto <ip-protocol>} {<ipv6-source-address/
    prefix>|any} {<ipv6-destination-address/prefix>|any} [log]

no [<sequence-number>]
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Specifies the packets to reject.
permit	Specifies the packets to accept.
ip	IP packet.
any	Any packet.
proto *<ip-protocol>*	The IP Protocol type specified by it protocol number <1-255>.
<ip-protocol>	The IP protocol number, as defined by IANA (Internet Assigned Numbers Authority **www.iana.org/assignments/protocol-numbers**).
	Protocol Number
	1
	2
	3
	4
	5
	6
	8

Parameter(cont.)	Description(cont.)
<ip-protocol> (cont.)	**Protocol Number**
	9
	11
	17
	20
	27
	28
	29
	30
	33
	48
	50
	51
	54
	58
	59
	60
	88
	89
	97
	98
	108
	112
	134
	135
	136
	137
	138
	139-252
	253
	254
	255
<ipv6-source-address/prefix>	IPv6 source address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
any	Any source address or local address.

Parameter(cont.)	Description(cont.)
<ipv6-destination-address/prefix>	IPv6 destination address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
`any`	Any destination address or remote address.
`log`	Log the results.

Mode IPv6 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage The filter entry will match on any IP protocol type packet that has the specified source and destination IPv6 addresses and the specified IP protocol type. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add a new ACL filter entry to the extended IPv6 access-list named `my-list` with sequence number `5` rejecting the IPv6 packet from `2001:db8:1:1` to `2001:db8:f:1`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 deny ip 2001:db8:1::1/128
2001:db8:f::1/128
```

To remove the ACL filter entry to the extended IPv6 access-list named `my-list` with sequence number `5`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# no 5
```

Related Commands
ipv6 access-list extended (named)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

(ipv6 access-list extended TCP UDP filter)

Use this ACL filter to add a filter entry for an IPv6 source and destination address and prefix, with a TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) source and destination port specified, to the current extended IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a filter entry for an IPv6 source and destination address and prefix, with a TCP or UDP source and destination port specified, from the current extended IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Syntax [tcp|udp]

```
[<sequence-number>] {deny|permit} {tcp|udp} {<ipv6-source-address/
   prefix>|any} {eq <sourceport>|lt <sourceport>|gt <sourceport>|
   ne <sourceport>} {<IPv6-destination-address/prefix>|any}
   {eq <destport>|lt <destport>|gt <destport>|ne <destport>} [log]

no {deny|permit} {tcp|udp} {<ipv6-source-address/prefix>|any}
   {eq <sourceport>|lt <sourceport>|gt <sourceport>|
   ne <sourceport>}} {<IPv6-destination-address/prefix>|any}
   {eq <destport>|lt <destport>|gt <destport>|ne <destport>} [log]

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Specifies the packets to reject.
permit	Specifies the packets to accept.
tcp	TCP packet.
udp	UDP packet.
<ipv6-source-address/prefix>	IPv6 source address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
any	Any source address or local address.
eq	Equal to.
lt	Less than.
gt	Greater than.
ne	Not equal to.
<sourceport>	The source port number, specified as an integer between 0 and 65535.
<ipv6-destination-address/prefix>	IPv6 destination address, or local address. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.

Parameter(cont.)	Description(cont.)
<destport>	The destination port number, specified as an integer between 0 and 65535.
log	Log the results.

Mode IPv6 Extended ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage The filter entry will match on any packet that has the specified source and destination IPv6 addresses and the specified TCP or UDP source and destination port. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add a new filter entry with sequence number `5` to the access-list named `my-list` to reject TCP packets from `2001:0db8::0/64` port `10` to `2001:0db8::f/64` port `20`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 deny tcp 2001:0db8::0/64 eq 10
2001:0db8::f/64 eq 20
```

To add a new filter entry with sequence number `5` to the extended IPv6 access-list named `my-list` to reject UDP packets from `2001:0db8::0/64` port `10` to `2001:0db8::f/64` port `20`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# 5 deny udp 2001:0db8::0/64 eq 10
2001:0db8::f/64 eq 20
```

To remove the filter entry with sequence number `5` to the extended IPv6 access-list named `my-list`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list extended my-list
awplus(config-ipv6-ext-acl)# no 5
```

Related Commands **ipv6 access-list extended (named)**
show ipv6 access-list (IPv6 Software ACLs)
show running-config

ipv6 access-list standard (named)

This command configures an IPv6 standard access-list for filtering frames that permit or deny IPv6 packets from a specific source IPv6 address.

The **no** variant of this command removes a specified IPv6 standard access-list.

Syntax [list-name]

```
ipv6 access-list standard <ipv6-acl-list-name>

no ipv6 access-list standard <ipv6-acl-list-name>
```

Parameter	Description
<ipv6-acl-list-name>	A user-defined name for the IPv6 software standard access-list.

Syntax [deny|permit]

```
ipv6 access-list standard <ipv6-acl-list-name> [{deny|permit}
   {<ipv6-source-address/prefix-length>|any} [exact-match]]

no ipv6 access-list standard <ipv6-acl-list-name> [{deny|permit}
   {<ipv6-source-address/prefix-length>|any} [exact-match]]
```

Parameter	Description
<ipv6-acl-list-name>	A user-defined name for the IPv6 software standard access-list.
`deny`	The IPv6 software standard access-list rejects packets that match the type, source, and destination filtering specified with this command.
`permit`	The IPv6 software standard access-list permits packets that match the type, source, and destination filtering specified with this command.
<ipv6-source-address/prefix-length>	Specifies a source address and prefix length. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64.
`any`	Matches any source IPv6 address.
`exact-match`	Exact match of the prefixes.

Mode Global Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage Use IPv6 standard access-lists to control the transmission of IPv6 packets on an interface, and restrict the content of routing updates. The switch stops checking the IPv6 standard access-list when a match is encountered.

For backwards compatibility you can either create IPv6 standard access-lists from within this command, or you can enter `ipv6 access-list standard` followed by only the IPv6 standard access-list name. This latter (and preferred) method moves you to the `(config-ipv6-std-acl)` prompt for the selected IPv6 standard access-list, and from here you can configure the filters for this selected IPv6 standard access-list.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Example To enter the IPv6 Standard ACL Configuration mode for the access-list named `my-list`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)#
```

Related Commands
(ipv6 access-list standard filter)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

(ipv6 access-list standard filter)

Use this ACL filter to add a filter entry for an IPv6 source address and prefix length to the current standard IPv6 access-list. If a sequence number is specified, the new entry is inserted at the specified location. Otherwise, the new entry is added at the end of the access-list.

The **no** variant of this command removes a filter entry for an IPv6 source address and prefix from the current standard IPv6 access-list. You can specify the filter entry for removal by entering either its sequence number, or its filter entry profile.

Syntax [icmp]

```
[<sequence-number>] {deny|permit} {<ipv6-source-address/
   prefix-length>|any}

no {deny|permit} {<ipv6-source-address/prefix-length>|any}

no <sequence-number>
```

Parameter	Description
<sequence-number>	<1-65535> The sequence number for the filter entry of the selected access control list.
deny	Specifies the packets to reject.
permit	Specifies the packets to accept.
<ipv6-source-address/prefix-length>	IPv6 source address and prefix-length in the form X:X::X:X/P.
any	Any IPv6 source host address.

Mode IPv6 Standard ACL Configuration

Default Any traffic controlled by a software ACL that does not explicitly match a filter is denied.

Usage The filter entry will match on any IPv6 packet that has the specified IPv6 source address and prefix length. The parameter `any` may be specified if an address does not matter.

Note Software ACLs will **deny** access unless **explicitly permitted** by an ACL action.

Examples To add an ACL filter entry with sequence number `5` that will deny any IPv6 packets to the standard IPv6 access-list named `my-list`, enter the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)# 5 deny any
```

To remove the ACL filter entry that will deny any IPv6 packets from the standard IPv6 access-list named `my-list`, enter the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)# no deny any
```

Alternately, to remove the ACL filter entry with sequence number `5` to the standard IPv6 access-list named `my-list`, enter the commands:

```
awplus# configure terminal
awplus(config)# ipv6 access-list standard my-list
awplus(config-ipv6-std-acl)# no 5
```

Related Commands

ipv6 access-list standard (named)
show ipv6 access-list (IPv6 Software ACLs)
show running-config

ipv6 prefix-list

Use this command to create an IPv6 prefix list or an entry in an existing prefix list.

Use the no variant of this command to delete a whole prefix list or a prefix list entry.

Syntax

```
ipv6 prefix-list <list-name> [seq <1-429496725>] {deny|permit} {any|
   <ipv6-prefix>} [ge <0-128>] [le <0-128>]

ipv6 prefix-list <list-name> description <text>

no ipv6 prefix-list <list-name> [seq <1-429496725>]

no ipv6 prefix-list <list-name> [description <text>]
```

Parameter	Description
`<list-name>`	Specifies the name of a prefix list.
`seq <1-429496725>`	Sequence number of the prefix list entry.
`deny`	Specifies that the prefixes are excluded from the list.
`permit`	Specifies that the prefixes are included in the list.
`<ipv6-prefix>`	Specifies the IPv6 prefix and prefix length in hexadecimal in the format X:X::X:X/M.
`any`	Any prefix match. Same as **::0/0 le 128**.
`ge <0-128>`	Specifies the minimum prefix length to be matched.
`le <0-128>`	Specifies the maximum prefix length to be matched.
`description`	Prefix list specific description.
`<text>`	Up to 80 characters of text description of the prefix list.

Mode Global Configuration

Usage When the device processes a prefix list, it starts to match prefixes from the top of the prefix list, and stops whenever a permit or deny occurs. To promote efficiency, use the seq parameter and place common permits or denials towards the top of the list. If you do not use the seq parameter, the sequence values are generated in a sequence of 5.

The parameters ge and le specify the range of the prefix lengths to be matched. The parameters ge and le are only used if an ip-prefix is stated. When setting these parameters, set the le value to be less than 128, and the ge value to be less than or equal to the le value and greater than the ip-prefix mask length.

Prefix lists implicitly exclude prefixes that are not explicitly permitted in the prefix list. This means if a prefix that is being checked against the prefix list reaches the end of the prefix list without matching a permit or deny, this prefix will be denied.

Example To check the first 32 bits of the prefix `2001:db8::` and the subnet mask must be greater than or equal to `34 and less than or equal to 40`, enter the following commands:

```
awplus# configure terminal

awplus(config)# ipv6 prefix-list mylist seq 12345 permit
2001:db8::/32 ge 34 le 40
```

Related Commands
match ipv6 address
show ipv6 prefix-list
show running-config ipv6 prefix-list

show ipv6 access-list (IPv6 Software ACLs)

Use the **show ipv6 access-list standard** command to display a specified standard named IPv6 access-list that has been defined using the **ipv6 access-list standard (named)** command.

Syntax

```
show ipv6 access-list standard <access-list-name>
```

Parameter	Description
`standard`	Named standard access-list.
`<access-list-name>`	Specify an IPv6 access-list name.

Mode User Exec and Privileged Exec

Example To show the ipv6 access-list specified with the name `acl_name` use the following command:

```
awplus# show ipv6 access-list standard acl_name
```

Output **Figure 61-1: Example output from the show ipv6 access-list standard command**

```
Named Standard IPv6 access-list name
  deny any
```

Related Commands
ipv6 access-list extended (named)
(ipv6 access-list extended IP protocol filter)
ipv6 access-list standard (named)
(ipv6 access-list extended TCP UDP filter)
(ipv6 access-list standard filter)

show ipv6 prefix-list

Use this command to display the prefix-list entries. Note that this command is valid for RIPng and BGP4+ routing protocols only.

Syntax `show ipv6 prefix-list [<name>|detail|summary]`

Parameter	Description
<name>	Specify the name of an individual IPv6 prefix list.
`detail`	Specify this parameter to show detailed output for all IPv6 prefix lists.
`summary`	Specify this parameter to show summary output for all IPv6 prefix lists.

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 prefix-list
awplus# show ipv6 prefix-list 10.10.0.98/8
awplus# show ipv6 prefix-list detail
```

Related Commands ipv6 prefix-list

Chapter 62: Quality of Service (QoS) Introduction

Introduction 62.2
QoS Operations 62.2
QoS Packet Information 62.3
Link Layer QoS 62.3
Differentiated Services Architecture 62.4
The Differential Services Field 62.5
Processing Pre-Marked Packets 62.6
Applying QoS on Your Switch 62.7
Classifying Your Data 62.7
Class Maps 62.7
Policy Maps 62.10
Premarking and Remarking Your Traffic 62.11
CoS to Egress Queue Premarking 62.11
DSCP to Egress Queue Premarking 62.13
Rate Limiting Traffic 62.14
Policing (Metering) Your Data 62.14
Single-Rate Three-Color Policing 62.15
Two-Rate Three-Color Policing 62.16
Configuring and Applying a Policer 62.17
Remarking Your Data 62.18
Configuring the Egress Queues 62.19
Egress Queues and QoS markers 62.19
Egress Queue Commands Hierarchy 62.19
Egress Queue Shaping 62.21
Scheduling 62.21
Drop Mode 62.22
Storm Protection 62.24
Policy-Based Routing 62.25
Practical Example 62.25

Introduction

This chapter introduces the concept of Quality of Service (QoS) with particular reference to Allied Telesis switches running the AlliedWare Plus™ Operating System.

The concept of QoS is a departure from the original networking concept of treating all network traffic in the same way. Without QoS, all traffic types are equally likely to be dropped when a link becomes oversubscribed. With QoS, certain traffic types can be given preferential treatment. QoS is therefore a very useful tool both to control congestion and to meter or cap data in order to apply pre-agreed service levels.

Operationally, QoS is applied within the link and network layers. Functionally it provides the capability to intelligently transport your network traffic in order to provide stable and predictable end-to-end network performance.

Business benefits

Quality of Service mechanisms enable:

- network service providers to sell different levels of service to customers, based on what their customers require, and be confident in their ability to guarantee the reliable delivery of these services
- enterprise and educational organizations to actively manage and provide many services across one network, for example live video streaming and standard data services, with preferential treatment being given to mission-critical traffic
- network administrators to manage network congestion as network traffic levels increase and time-critical applications, such as streaming media, become more widely in demand by customers and organizations

QoS Operations

Quality of Service is typically based on how the switch performs the following functions:

- assigns priority to incoming frames (that do not already carry priority information)
- correlates prioritized frames with traffic classes, or maps frames to traffic classes based on other criteria
- correlates traffic classes with egress queues, or maps prioritized frames to egress queues
- provides minimum and maximum bandwidths for traffic classes, egress queues, and/or ports
- schedules frames in egress queues for transmission (for example, empty queues in strict priority or sample each queue)
- re-labels the priority of outgoing frames
- determines which frames to drop or re-queue if the network becomes congested
- reserves memory for switching/routing or QoS operation (for example, reserving buffers for egress queues or buffers to store packets with particular characteristics)

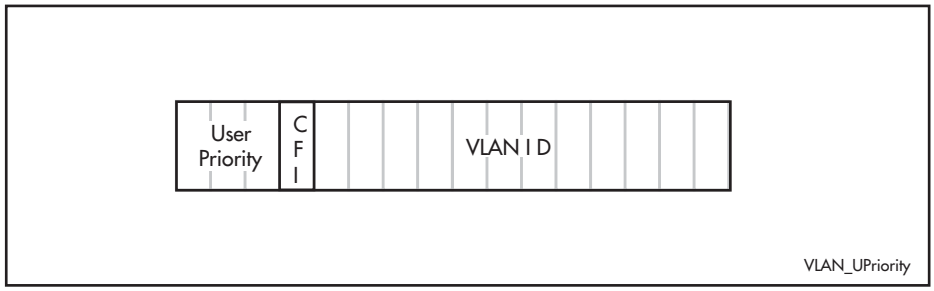
QoS Packet Information

Provision for QoS information to be embedded within the data fields exists within both the data link and network layer protocols. This information can then be used to assess the priority of the data and the resource preferences that need to be applied. The process of applying these service quality tags to your data is known as marking.

Link Layer QoS

Link layer frames entering a port may either be tagged or untagged. VLAN tagged frames contain the additional 802.1Q tag fields shown in **Figure 62-1** below. Located within the TCI is a three bit User Priority field. This field is specifically provided to attach QoS based priority information, often referred to as the Class of Service (CoS) field.

Figure 62-1: IEEE 802.1Q tagging



Appendix G of the IEEE Standard 802.1D provides some useful guidelines on applying priorities to 7 traffic types. These are summarized in the **Table 62-1** below.

Table 62-1: CoS traffic mapping guidelines

User Priority	Traffic Types
1	Background
2	Spare
0	Best Effort
3	Excellent Effort
4	Controlled Load
5	Video <100 ms latency and jitter
6	Voice <10 ms latency and jitter
7	Network Control

On the switch you can use the **match cos** command to select frames that match a particular User Priority value and assign them to a particular class-map. You can then map these incoming frames to an egress queue. This facility enables you to accept frames that are already carrying meaningful priority information and automatically assign them to an appropriate egress queue. For example, you could decide to send frames with a User Priority value of 2 to queue 0. The process of assigning queues based on CoS tags is commonly known as “Pre-marking”.

Note

You configure the pre-marking steps on an ingress port. This process marks the data packets so that when they reach the egress port the decisions made during pre-marking can be applied in accordance with the configuration of the egress port.

Application with VLAN double tagging

Note that if you are using VLAN double tagging, you could use the **match cos** command to set the individual QoS requirements *within* each client network and also separately within the provider network. You can then use the **match inner-cos** command to apply particular *client* QoS requirements that you want to apply within the provider network. This process applies two levels of QoS within the provider network; one that operates specifically for the network provider, and another that is specific for traffic belonging to selected clients. See **"VLAN Double Tagging (VLAN Stacking)" on page 18.5**.

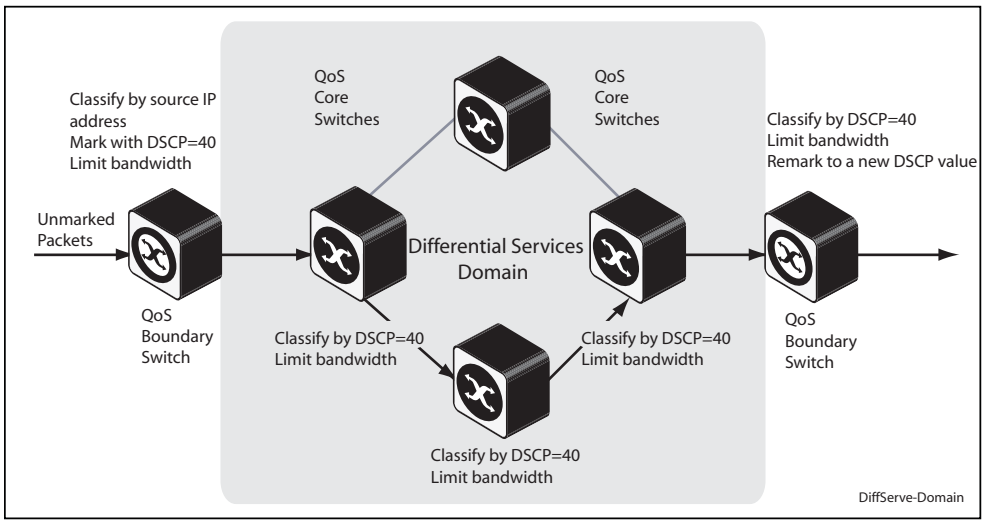
At the network layer IPv4 packets contain an 8 bit field specifically to carry QoS information. This field, defined in RFC 1349, was originally named the Type of Service (ToS) field and contained a *ToS* component and a *Precedence* component. The ToS field however, has since been replaced by the Differentiated Services field.

Differentiated Services Architecture

Whilst a full description of the differential services model is outside the scope of this software reference, a brief introduction is provided. For further information, RFC 2475 provides an in depth definition of the architecture.

The basic differential services model envisages a multi router network within which common service qualities are applied. At the network boundary, *QoS Edge Routers* inspect the traffic and classify it into common service quality groups called Per Hop Behaviors (PHBs). A specific marker value called a Differential Services Code Point (DSCP) is added to the IP header of each packet, which allocates it to a PHB. *QoS Core Routers* within the network can then use the DSCP to decide on an appropriate service quality level to apply. When a network contains a consistently applied differential services code points DSCP it is referred as a Differential Services Domain (often shortened to DiffServe Domain). **Figure 62-2** shows a simple Differential Services Domain.

Figure 62-2: Differentiated Services Domain

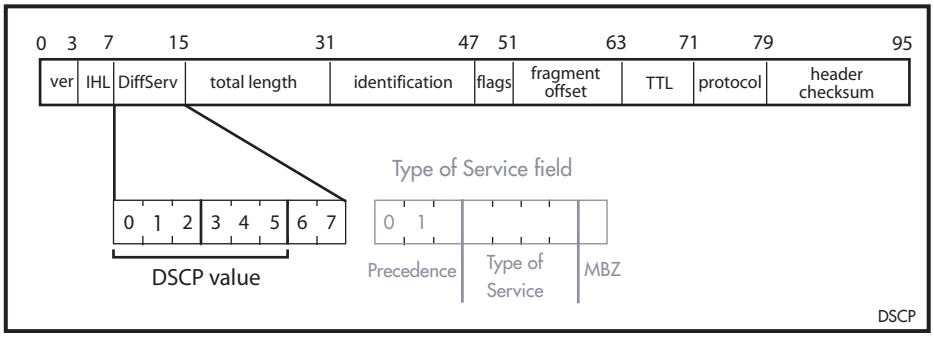


The Differential Services Field

Figure 62-3 shows an IP header containing a Differentiated Services field. The format of this redefined field is explained in RFC 2474; the main difference being that the old ToS field has been replaced by a 6 byte Differentiated Services Code Point (DSCP) field, which now provides for up to 64 defined values.

By applying this model only the QoS edge routers need to fully interrogate the incoming data packets; the QoS core routers are then relieved of this processing task and need only to inspect the DCSP before applying its appropriate forwarding, queuing, and shaping rules.

Figure 62-3: The DSCP bits of the DS field in the IPv4 header



On the switch you can use the **match dscp** command to select frames containing a particular DSCP value, and associate them with a particular class-map and policy-map.

Because the model offers considerable flexibility, and the mapping of traffic types to DCSPs is individual for each network, this locally applied definition is known as a *Differential Services Domain*. The previous section introduced the concept of a Per Hop (service quality) Behaviors or PHBs. RFC 2597 defines a specific PHB group called Assured Forwarding (AF). The AF PHB group provides delivery of IP packets in four independently forwarded AF classes. Within each AF class, an IP packet can be assigned one of three different levels of drop precedence. **Table 62-2** shows a list of recommended AF code points.

Table 62-2: Recommended DSCP code points

	(Lowest Priority) Class 1 (001xxxx)	Class 2 (010xxxx)	Class 3 (011xxxx)	(Highest Priority) Class 4 (100xxxx)
Low Drop Precedence	001010 AF11 Decimal 10	010010 AF21 Decimal 18	011010 AF31 Decimal 26	100010 AF41 Decimal 34
Medium Drop Precedence	001100 AF12 Decimal 12	010100 AF22 Decimal 20	011100 AF32 Decimal 28	100100 AF42 Decimal 36
High Drop Precedence	001110 AF13 Decimal 14	010110 AF23 Decimal 22	011110 AF33 Decimal 30	100110 AF43 Decimal 38

Processing Pre-Marked Packets

A logical question to ask at this point is: how does the QoS switch deal with data that arrives with a pre-existing service level tag such as a DSCP? As previously touched on, the differentiated services model envisages a network that comprises QoS boundary routers at its edge and QoS core routers in its core network.

At the network edge the QoS boundary routers filter the incoming data based on specific packet components. Based on this filtering each packet is assigned a DSCP value. This value will determine the service level (priority and queuing, for example) that will be applied.

Within the network core, the packet filtering required is reduced to simply reading the DSCP within each incoming packet, and applying the appropriate set of service levels. This relieves the core routers of the processing overhead of applying complex filtering to its high speed data streams.

Applying QoS on Your Switch

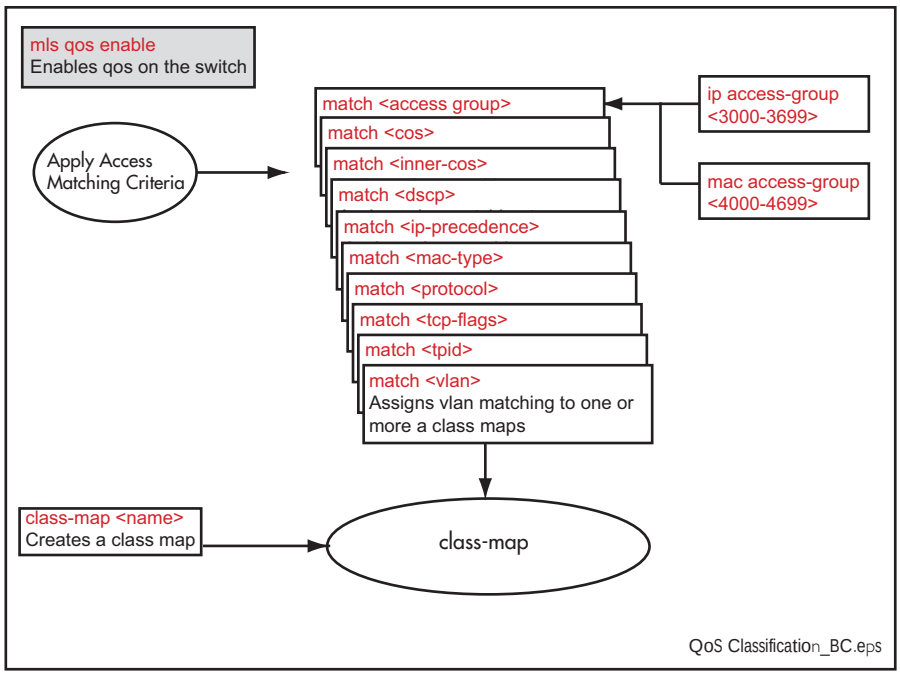
This section steps you through the various stages of QoS set-up and introduces the QoS commands and how to apply them.

Before you configure any QoS functions on your switch, first enable QoS by using the **mls qos enable** command.

Classifying Your Data

One of the early steps in setting up QoS on a network is planning and applying your classification rules. Classification is the process of **Filtering** and **Marking**. Filtering involves sorting your data into appropriate traffic types. Marking involves tagging the data so that downstream ports and routers can apply appropriate service policy rules. **Figure 62-4** illustrates the classifying process, and will be referred to in the examples that follow.

Figure 62-4: QoS classification process



At the premarking stage you can assign your data a particular priority level by giving it a link level user priority, see **"Link Layer QoS" on page 62.3**, or a network level DSCP **"Differentiated Services Architecture" on page 62.4**. You can also assign the data to a particular output (or egress) queue.

Class Maps

Class Maps are among the pivotal QoS components. They provide the means that associate the classified traffic with its appropriate QoS actions. They are the linking elements for the following functions:

- classification
- policy-mapping
- pre-marking

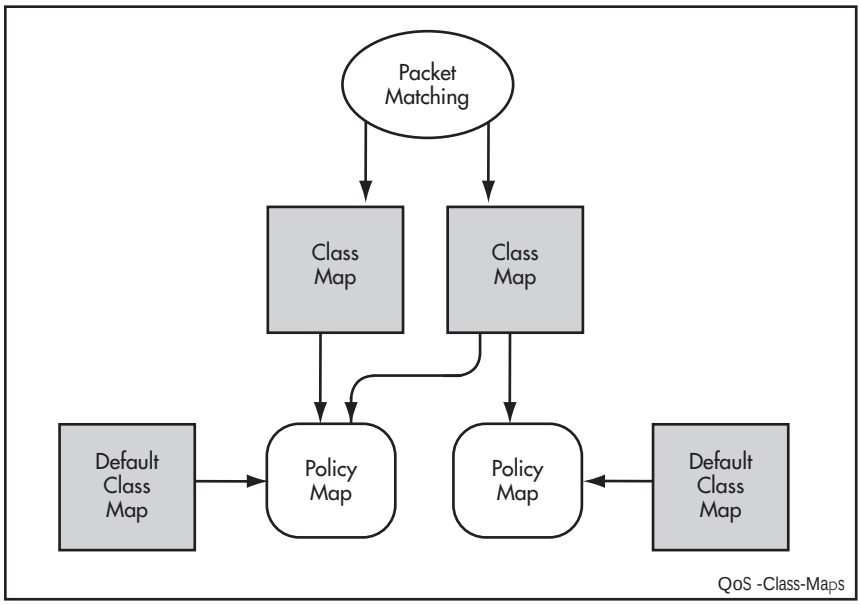
Figure 62-5 shows the relationship between a class-map and its associated functions. Note that the relationship between a class-map and a policy-map can be one-to-one or many-to-one. For information on policy-maps see the section, **"Policy Maps" on page 62.10**.

Note

If a conflict occurs between the settings in two class-maps, priority will be applied to the class-map that was created first.

An example of such a conflict is the arrival of a packet that meets the classification requirements of two class-maps each configured to the same policy-map and set to apply different priority settings to the packet.

Figure 62-5: Relationship between a class-map and its associated functions



Creating a class-map

To create a class-map, use the **class-map** command on page 63.3.

This example creates a class-map called **video-traffic** and another called **data-traffic**:

```
awplus# configure terminal
awplus(config)# class-map video-traffic
awplus(config-cmap)# exit
awplus(config)# class-map data-traffic
awplus(config-cmap)#
```

Creating and configuring default class-maps

These (automatically created) default class-maps serve as the means to specify the action that will apply to all unclassified data, i.e. all data within a policy-map that is not captured by any of the applied match commands that are applied to the policy-map by its class-maps.

Each time a new policy-map is created a new class-map called "default" is also automatically created and assigned to the new policy-map. You can configure any of the default class-maps by using the **default-action** command on page 63.5

To set the default class-map for the policy-map **p-map1** to have the action of `deny`:

```
awplus# config
awplus(config)# policy-map p-map1
awplus(config-pmap)# default-action deny
```

Applying a match command to a class-map

To apply a matching filter to a class-map use one of the match commands.

This example creates a filter to select VLAN 5 traffic and applies this filter to the class-map named **video-traffic**.

```
awplus# config terminal
awplus(config)# class-map video-traffic
awplus(config-cmap)# match vlan 5
```

Associating a class-map with a policy-map

To associate a class-map with a policy-map, use the **class** command on page 63.2.

Note A maximum of 128 class-maps may be attached to each policy-map.

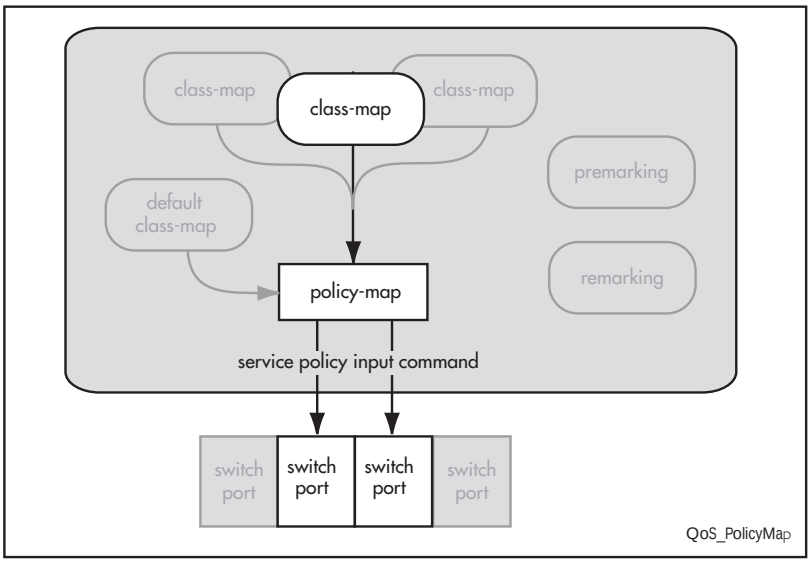
The following example creates a policy-map called **policy-one**, and associates it with the class-maps named **video-traffic**, and **database-traffic**:

```
awplus# configure terminal
awplus(config)# policy-map policy-one
awplus(config-pmap)# class video-traffic
awplus(config-pmap-c)# exit
awplus(config-pmap)# class database-traffic
awplus(config-pmap-c)#
```

Policy Maps

Policy maps are the means by which you apply your class-map properties to physical switch ports. **Figure 62-6 on page 62.10** illustrates this concept. Note that a policy-map can be assigned to several ports, but a port cannot have more than one policy-map assigned to it.

Figure 62-6: Policy maps and related entities



To create and name a new policy-map you use the **policy-map** command on page 63.30.

To create a policy-map called `pmap1` use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
```

Having created the policy-map `pmap1` we can use the **class** command on page 63.2 to assign it to one or more class-maps. Since we created the class-maps `video-traffic` and `office-traffic` earlier in this chapter, we can now attach the policy-map `pmap1` to both class-maps.

Use the **class** command to assign the policy-map `pmap1` to the class-maps `video-traffic` and `office-traffic`:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class video-traffic
awplus(config-pmap-c)# exit
awplus(config-pmap)# class office-traffic
awplus(config-pmap-c)#
```

Premarking and Remarking Your Traffic

Premarking relates to adding QoS markers to your incoming data traffic before it is metered (policed). Remarking is the same process when applied after metering. Network switches will often be configured with two different premarking profiles, one for the QoS edge switches and another for the QoS core switches. This situation would apply if you are operating DSCP domains.

QoS markers can be applied at both the link layer (within the CoS field), and at the network layer (within the DSCP field). For more information on this topic see **"QoS Packet Information" on page 62.3**.

For boundary QoS switches

Traffic entering QoS boundary switches is unlikely to contain pre-existing QoS tagging. In this case, you can:

- Assign a CoS tag to data associated with a particular class-map.
- Use the **trust dscp** command to enable the mls qos map premark DSCP map. This map enables you to change the DSCP tag and also map the tag to an egress port queue, a CoS value, or both. At the premarking stage you can set this mapping using the command **mls qos map premark-dscp to**. After policing, you can then use the **remark-map** command to change the DCSP based on the packet's bandwidth class, or remap the existing bandwidth class, to a new value.

For an untagged packet, if no other mapping is applied and the packet is untagged, (i.e. in the absence of any other queue selection) traffic will be sent to queue 2.

For core QoS switches

Traffic entering ports within the QoS core network will almost certainly contain some pre-existing QoS tagging. Where this is the case, you can:

- Map the CoS tag to an egress queue. You can do this either for the whole switch or for specific ports via their assigned policy-maps. See **"CoS to Egress Queue Premarking" on page 62.11**.
- Map the DSCP tag to an output queue. You can do this either for the whole switch or for specific ports via their assigned policy-maps.
- Remap incoming data DSCP or CoS tags to values that are more appropriate for a particular switch or network.
- Assign bandwidth classes for your packets, based on the incoming DSCP. See **"DSCP to egress queue premarking commands" on page 62.13**.

CoS to Egress Queue Premarking

If you are using CoS tagging for your QoS functions, your traffic is likely to be either entering the switch with a pre-existing CoS tag, or will have appropriate tags attached via your class-maps and policy-maps. You can now mark the data for a particular egress queue, which will take effect when the data reaches its output port. There are two fundamental methods of applying CoS tagged packets to egress queues:

1. Apply a global mapping of CoS tags to egress queues for all ports.
2. Apply a CoS to egress queue mapping for the class-map / policy-map. This mapping - which forms part of the policy-map - is applied at an input port, but will take effect at the packet's destination output port. Note that this procedure takes priority over that described in method (1) above.

These methods and their related commands will be now be described in greater detail.

Note Where a packet contains both a CoS and a DSCP field, and each field maps to a different class-map; the switch will apply a priority that is based on the date that the class-map was added to the policy-map; the earlier the date, the higher the priority.

Mapping CoS tags to traffic types

The command **mls qos map cos-queue to** enables you to create a switch-wide mapping of CoS values to egress queues. The default mappings for this command are:

```
COS :           0 1 2 3 4 5 6 7
--------------------------------
QUEUE:          2 0 1 3 4 5 6 7
```

These mappings match the CoS guidelines documented in Annex H.2 of ANSI/IEEE 802.1D 1988 Edition. Table H-15 on page 355 of the standard shows a table of user priorities for specific traffic types. **Table 62-3** shows an adapted version of the ANSI/IEEE table.

Table 62-3: Traffic type guidelines

User Priority (egress queue)	CoS Value	Acronym	Traffic type	Internal Traffic Queue Defaults
0 (lowest)	1	BK	Background	
1	2	-	Spare	
2	0	BE	Best Effort	Default
3	3	EE	Excellent Effort	
4	4	CL	Controlled Load	
5	5	VI	"Video," <100 ms latency and jitter	
6	6	VO	"Voice," <10 ms latency and jitter	EPSR-Management BPDU ARP-Requests
7 (highest)	7	NC	Network Control	Stack Management

CoS settings for VCStack stack operation

In general you can apply the same principles when configuring QoS on a VCStack as you would for single switch; however there are a few specific changes that you will need to make.

Switches within a VCStack exchange their stack management information and user data over their high speed inter-stacking links. The stack management information is pre-assigned to the egress queue 7. This is the highest value queue, and (in a stacked configuration) its traffic should not be shared with any user data. However, any CoS tagging of 7 applied to the incoming data will automatically be assigned to queue 7 as it crosses the internal stacking links. You will therefore need to reconfigure your CoS to Queue settings to ensure that no user data is sent to queue 7.

To prevent this from happening, we recommend that you make appropriate changes to your queue settings (mappings) to reflect the stacking requirement previously described. For more information on this topic, see **"Mapping CoS tags to traffic types" on page 62.12**.

This process should include (but not be limited to) running the following command to ensure that any remaining user still carrying a CoS 7 tag will be mapped to egress queue 6.

To remap priority CoS traffic to egress queue 6, run the following command.

```
awplus# config terminal
awplus(config)# mls qos map cos-queue 7 to 6
```

DSCP to Egress Queue Premarking

If you are using DSCP tagging for your QoS functions, your traffic is likely to be entering the switch either with a pre-existing DSCP tag, or will have appropriate DSCP tags attached via your class-maps and policy-maps. You can now mark the data for a particular egress queue, which will take effect when the data reaches its output port.

If your switch forms part of a DSCP domain, you can adapt the steps in this section to apply the mappings and settings to match the standards you have selected for the domain. This mapping - which forms part of the policy-map - is applied at an input port, but will take effect at the packet's destination output port.

DSCP to egress queue premarking commands

A number of commands can be used for mapping DSCP tags. Where these conflict, the switch applies a pre-defined set of priorities. **Table 62-4** lists these priorities in order (lowest priority first).

Where a packet that contains both CoS and a DSCP fields and each field maps to a different class-map / policy-map, the switch will apply a priority based on the creation date of class-maps - the earlier the creation date, the higher the priority priorities.

Table 62-4: DSCP mapping commands in hierarchical order

Command	Function
trust dscp	Setting the trust dscp enables the **mls qos map premark-dscp to** command to apply. See **"Setting the Trust DSCP map" on page 62.14**.
mls qos map premark-dscp to	With the trust dscp set, this command applies a remapping table whose values include the dscp and egress queues.

Setting the Trust DSCP map

The Trust DSCP mapping table assigns a new set of QoS values for a DSCP value supplied as table input. To configure this table you use the **mls qos map premark-dscp to** command.

Table 62-5: Drop probability table

Table Input	---------- Table Output ----------			
Existing DSCP	New DSCP Value	New CoS Value	New Queue No	New BW Class green yellow red

The Trust DSCP map provides the highest priority of all the pre-marking controls. To apply this table you must first apply the trust setting by using the **trust dscp** command.

Rate Limiting Traffic

You can specify a rate limit for each port, and drop packets that exceed that rate. To do this, use the QoS policer function.

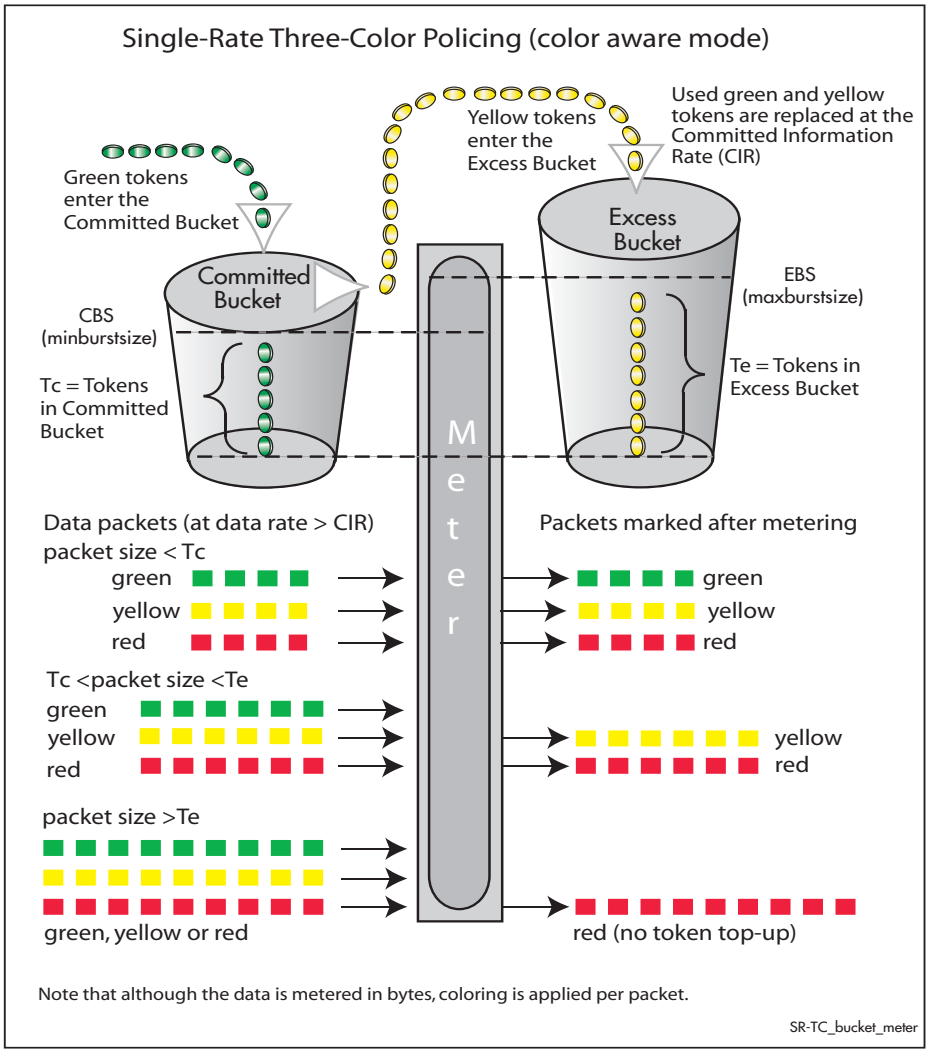
Policing (Metering) Your Data

Once you have set-up your classification and created your class-maps, you can start conditioning your traffic flows. One tool used for traffic conditioning is the policer (or meter). The principle of policing is to measure the data flow that matches the definitions for a particular class-map; then, by selecting appropriate data rates, allocate the flows into one of three categories: Red, Yellow, or Green. You then decide what action to apply to the colored data.

Single-Rate Three-Color Policing

This policing method is based on that defined in RFC 2697. The principle of single-rate three-color policing is shown in **Figure 62-7**. For a given class-map, a meter monitors both the token count in the buckets, and the input data flow.

Figure 62-7: Single-rate three-color policing



Each byte entering the meter is paired with a token in one of the buckets, and a token is removed as each byte is accepted. If the input data rate is the same as the CIR then the data passes through the port at the same rate as its bucket fills. Hence the bucket level remains constant. In this model, the data buffer is represented by two data buckets. You can specify the CIR using the **police single-rate action** command.

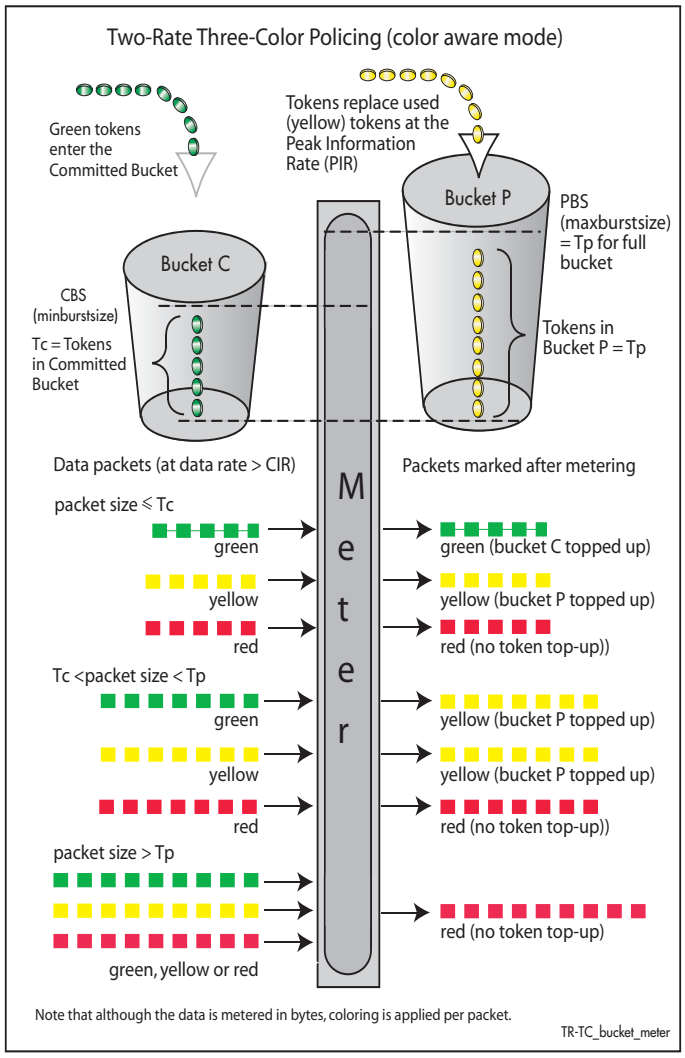
Initially both buckets have their full token count. A surge of data exceeding the CIR will begin to empty the bucket. As the data and tokens are paired, data bytes that match tokens below the CBS level are marked green, those that are between CBS and EBS will be marked yellow, and those that are above EBS are marked red.

Note that although the data is metered per byte, the color marking process is applied per packet. This means that if there were only sufficient tokens available to match part of a packet, then the whole packet would be marked red. Then, depending on the **action** parameter of the **police single-rate action** command, the whole packet will be either dropped or forwarded. In either situation, the red-marked packet will leave the bucket counts unchanged.

Two-Rate Three-Color Policing

This policing method is based on that defined in RFC 2698. The principle of two-rate three-color policing is shown in **Figure 62-8**.

Figure 62-8: Two-rate three-color policing



For a given class-map, the meter monitors the token count in both buckets, and the input data flow. Initially tokens enter both buckets until full. As the data enters a port, the meter pairs each byte to a token in one of the buckets, then removes a token from the appropriate bucket. Bucket C is topped up with tokens at the Committed Information Rate (CIR), and bucket P is topped up at the Peak Information Rate (PIR).

When data enters the port at the CIR, the bucket fills at the same rate as the incoming data, thus the token count in bucket C remains constant. Similarly, if data enters the port at the PIR, then the token count in bucket P remains constant. You can specify the CIR and the PIR by using the **police twin-rate action** command. The function of this command is explained in the section **"Configuring and Applying a Policer" on page 62.17**.

A surge of data exceeding the CIR will begin to empty bucket C. If bucket C empties to a point where it has insufficient tokens to match to an incoming data packet, then the data packet will be marked yellow. The data will now be measured against the level in bucket P and tokens will be removed from this bucket to match the incoming data. If the incoming data rate drops to less than the CIR then the data will continue to be marked yellow until the level in bucket C has had a chance to fill, whereupon it will be marked *green*.

If the incoming data is greater than the PIR, then bucket P begins to empty. If bucket P empties to a point where it has insufficient tokens to match to an incoming data packet, then the data packet will be marked *red*. In this situation no tokens are removed from either bucket.

Note that although the data is metered per byte, the color marking process is applied per packet. This means that if there were only sufficient tokens available to match part of a packet, then the whole packet would be marked red. Then, depending on the **action** parameter of the **police twin-rate action** command, the whole packet will be either dropped, or marked and forwarded. In either situation, the red marked packet will leave the bucket counts unchanged.

Configuring and Applying a Policer

The previous section showed how the policer works and how to select either the single rate or twin rate action. To apply a policy to class-maps:

- Select your policy-map and class-map from the command prompt, then enter either the **police single-rate action** command or the **police twin-rate action** command whilst selecting the appropriate command parameters.

This will apply the command to the selected class-map. By running this command several times, each for a different class-map, you can apply separate meter settings to each class-map.

Remarking Your Data

The remarking process enables you to change the QoS tagging and queue assignments from data that has already been marked by the policer. To do this you fill entries in the remarking table by using the **remark-map** command on page 63.32. In order to remark your data ensure that the **action** parameter of either the **police single-rate action** or the **police twin-rate action** is set to **remark-transmit**.

The following table shows the remarking options.

Table 62-6: Remarking table

BANDWIDTH CLASS		
Green	New DSCP	New bandwidth class (Red, Yellow, or Green)
Yellow	New DSCP	New bandwidth class (Red, Yellow, or Green)
Red	New DSCP	New bandwidth class (Red, Yellow, or Green)

Example Traffic presently marked Yellow is to be assigned a new DSCP value of 20:

Table 62-7: Remarking table example

BANDWIDTH CLASS		
Yellow	New DSCP = 20	Bandwidth class unchanged

To configure this setting, you would enter the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark-map bandwidth-class yellow to
new-dscp 20
```

Further remarking can be achieved by using the **remark new-cos** command on page 63.34. This command enables you to configure and remark either or both the CoS flag in the data packet, and the input into the CoS to queue map thus changing the destination egress queue.

Configuring the Egress Queues

Previous sections have explained the ingress functions. These include how the incoming data can be classified and marked according to its priority and allocated to an egress queue, then finally how metering and remarking is applied. At this point the data then flows across the switch to its destination egress port where its transit to the egress queues is controlled.

The means by which data is applied to the egress queues is dependent on three functions:

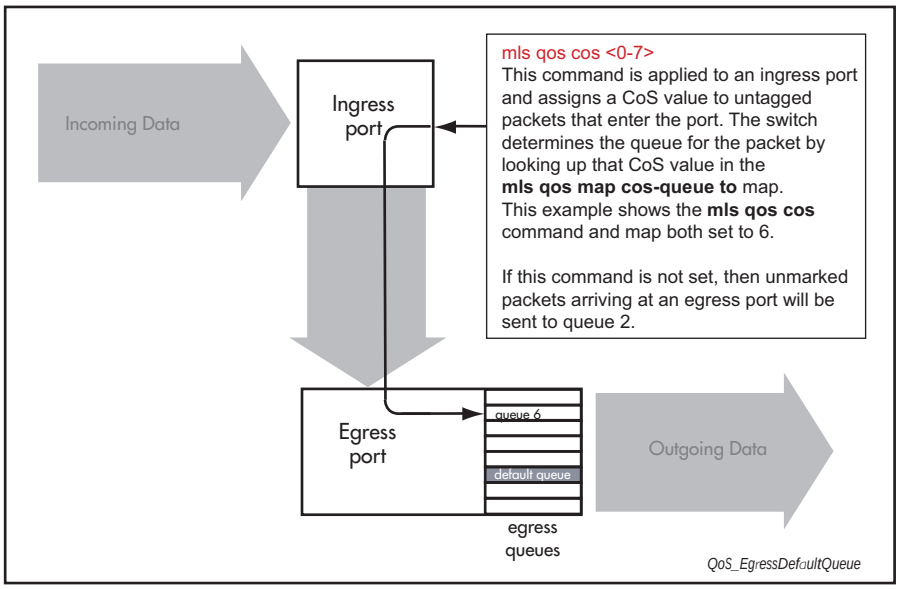
- Egress queue and QoS markers that are set within each data packet
- Egress controls that are applied to the whole switch
- Egress controls that are applied to each individual switch port

Egress Queues and QoS markers

Once the data packets have been appropriately filtered, classified, policed, and remarked, they travel across the switch's internal paths carrying their assigned QoS tag markers such as their priority, class and destination queues. For more details on ingress data marking, refer to the earlier sections of this chapter. At the egress port these markers are read and used to determine which queues each data packet will be forwarded to, and the priorities that will be applied.

There are eight egress queues allocated to each egress port. The egress queue that a particular packet passes through is determined by either the configuration of the switch, or by the markers contained within the packet itself.

Figure 62-9: Default egress queue



Egress Queue Commands Hierarchy

The destination queue that any one packet will take depends on the markers within the packet, and the way the queuing commands have been set. Also, some queuing commands will override others. Here is how the switch prioritizes its queuing commands.

Imagine a packet entering an ingress port then traveling through the switch fabric to reach its appropriate egress port. In this situation the following hierarchy will apply:

1. If the packet enters an egress port carrying no QoS markers and no QoS queuing commands have been set on the switch, then the packet will exit the port via queue number 2.

2. This default situation can be overridden by the **mls qos cos** command, which assigns a CoS value to untagged packets that enter a port. The switch determines the queue for the packet by looking up that CoS value in the **mls qos map cos-queue to** map.

3. If the packet contains a CoS value and arrives at an egress port with no other configuration applying, the switch determines the queue for the packet by looking up that CoS value in the **mls qos map cos-queue to** map.

4. The above CoS and queue settings can be overridden by the **remark new-cos** command. This command lets you change the CoS value for packets that match a given class-map, and therefore lets you change the queue for those packets.

Egress Queue Shaping

This section is concerned with how the egress queues are cleared.

Scheduling

The scheduler determines how packets in the egress port queues are serviced. Two servicing methods can be applied:

- strict priority
- weighted round robin (WRR)

Strict priority servicing

By default, all queues on all ports are serviced in a strict priority order. This means that the highest numbered priority queue (queue 7) is emptied first; then when it is completely empty, the next highest priority queue is processed, and so on. Thus, for a strict priority queue to be processed, all higher priority queues must be empty.

Strict priority servicing is the default setting; however if your system is configured for weighted round robin (WRR), you can return it to priority queuing by using the commands shown in the following example.

To return queue 2 of `port1.0.1` from WRR servicing to strict priority queuing, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# priority-queue 2
```

Weighted round robin servicing

The following examples show how to configure round robin servicing.

Example To configure a wrr-queue by applying a weighting value of 6 to queues 0 and1:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue weight 6 queues 0 1
```

Example In this example port 1.0.1 has queues configured as follows:

- queues 6 and 7 are configured strict priority
- queues 3 and 4 are configured as WRR with weighting values of 6
- queue 5 is configured as WRR with weighting values of 12
- queues 0, 1 and 2 are configured as WRR with weighting values of 4

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# priority-queue 6 7
awplus(config-if)# wrr-queue weight 6 queues 3 4
awplus(config-if)# wrr-queue weight 12 queues 5
```

In this example, the queues are processed as follows:

1. Queue 7 is processed first.
2. If queue 7 is empty, Queue 6 is processed next.
3. If queues 6 and 7 are empty, queue 5 is processed next.
4. If queue 5 is empty, queues 3 and 4 are processed with equal weighting.

Drop Mode

The drop mode sets the limits for packets in the egress queues and determines how packets will be dropped if the queues become congested. The switch supports the tail drop mode and is pre-configured with the following settings:

Data packets will be dropped per color at the following buffer usage:

Red at 60%, Yellow at 80%, and Green at 100%.

These settings cannot be reconfigured.

Tail drop

In this drop mode each egress queue is configured with a maximum threshold value. This value represents the point where the egress buffer queues are full and the egress port must start dropping data. The port does this by dropping data packets destined for the full queue on a “last in first dropped” basis. This enables the port to clear its data already queued for egress.

If a reliable transport protocol, such as TCP is used, this data should be retransmitted, but at a slower rate due to lack of returning “acknowledgments”.

Storm Protection

Storm protection uses QoS mechanisms to classify on traffic likely to cause a packet storm (broadcast and multicast). Unless you are running an enhanced storm protection feature such as Loop Protection, the per-port storm protection mechanism simply discards any traffic over the configured limit. However, with QoS storm protection, several actions are possible when a storm is detected:

- You can disable the port physically.
- You can disable the port logically.
- You can disable the port for a particular VLAN.

To enable the policy-based storm protection, use the **storm-protection** command on page 63.50.

Storm protection is activated as soon as a port is enabled, before the port forwards frames.

When a storm is detected on a port, a message is automatically recorded in the log, and you can configure an SNMP trap to signal that a port has been disabled. When a storm is detected on a trunk or port group, the entire trunk or port group is disabled.

The following table explains the basic concepts involved with storm protection.

Concept	Description
Window	The frequency at which traffic is measured to determine whether storm protection should be activated.
Rate	The amount of traffic per second that must be exceeded before the switch takes the configured action.
Action	What the switch does when it detects a storm on a port.
Timeout	The length of time the port remains disabled after a port has been disabled due to a packet storm.

To set the action to take when triggered by QoS Storm Protection (QSP), use the **storm-action** command on page 63.48.

To set the time to re-enable the port once disabled by QSP, use the **storm-downtime** command on page 63.49.

Policy-Based Routing

Policy based routing provides a means to create multiple paths to the same destination. The specific path that any particular packet will take can be based on configurable network metrics such as priority, protocol, or VLAN membership. For example, policy based routing can implement policies to allow or deny paths based on the identity of user devices, application, or packet sizes.

Practical Example

The example shown makes use of policy based routing to achieve the following:

1. Ensure that traffic being sent between local VLANs is switched normally.

2. Selects a particular egress path for traffic destined for the wide are networks.

Configuration overview

A large government building houses employees from three separate government departments: Heath, Welfare, and Employment. Each department has its own local subnet, and an associated VLAN; these are:

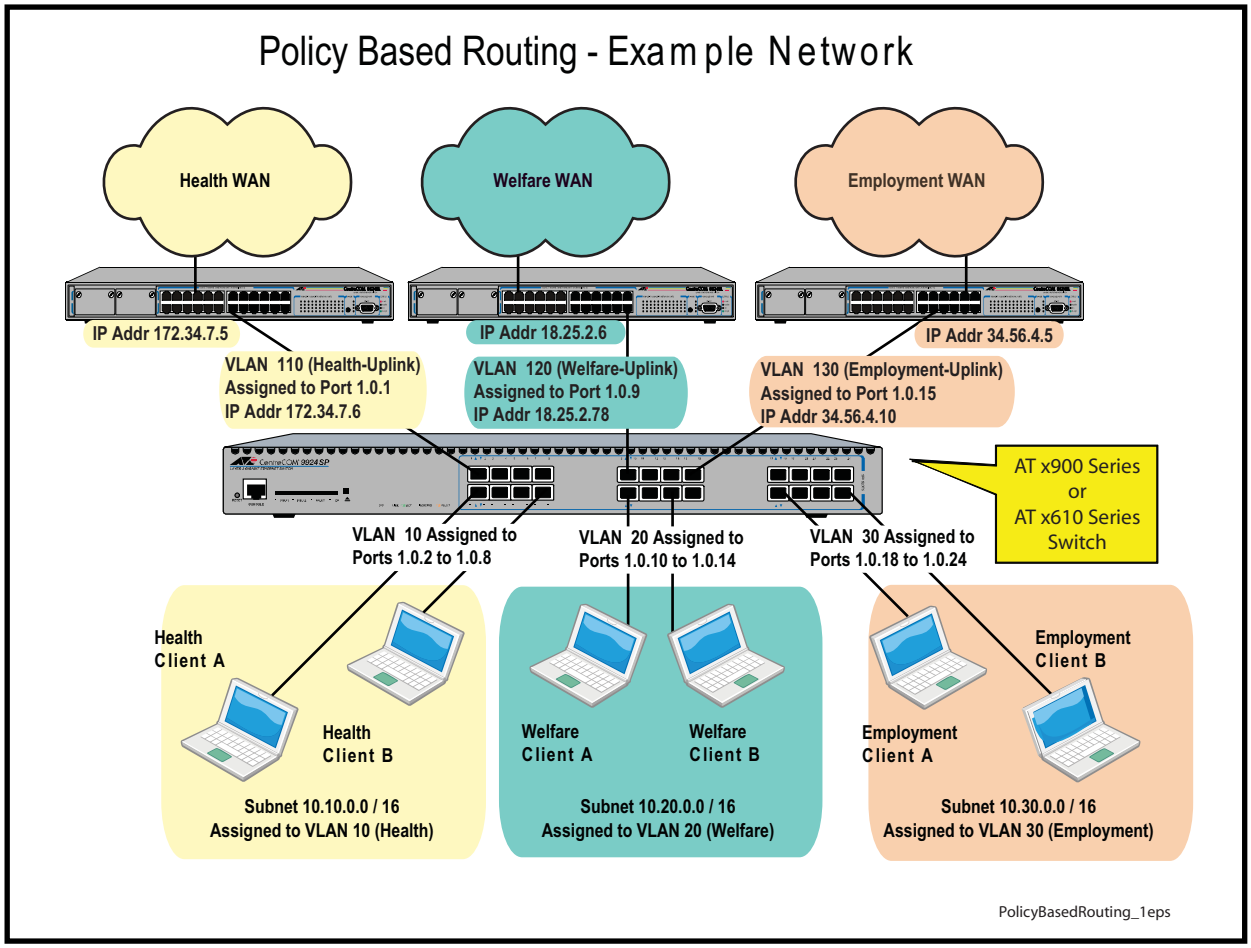
- 10.10.0.0/16 Health, VLAN 10
- 10.20.0.0/16 Welfare, VLAN 20
- 10.30.0.0/16 Employment, VLAN 30

Inquiries to each department are fed through a common Allied Telesis switch. The switch has 3 uplink ports, each of which (for simplicity) will be in a different VLAN and each will supply a connection to its relevant government department and to the Internet via each departments particular ISP (Internet Service Provider). These are:

- Port 1.0.1 Health Uplink, VLAN 110
- Port 1.0.5 Welfare Uplink, VLAN 120
- Port 1.0.21 Employment Uplink, VLAN 130

This configuration is illustrated in **Figure 62-10**.

Figure 62-10: Policy-based routing example



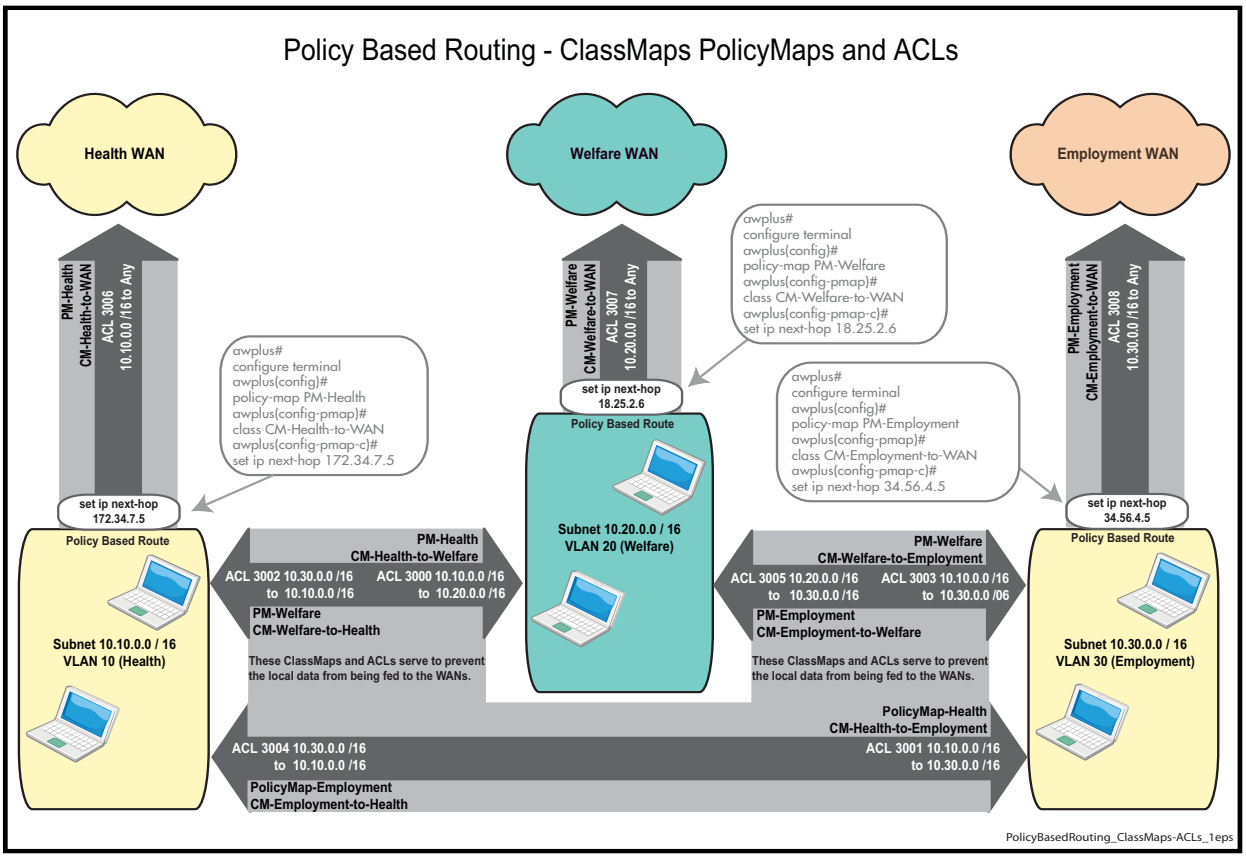
Configuration steps

The following steps can be used to setup this example network. Since each step involves entering several instances of a command type, a single practical command entry is shown at the end of each step. The set of steps comprise the following:

1. Create VLANs on the switch.
2. Create access control lists (ACLs) that will match the data flows between local subnets.
3. Create ACLs that will match the data flows between local devices and other destinations.
4. Setup class-maps for each department and apply an access-list to each of the class-maps.
5. Setup class-maps for each department's wide area connection and apply an access-list to each of these class-maps.
6. Create the departmental policy-maps and associate them with their appropriate class-maps.
7. Apply these policy-maps to their appropriate ports.

These class-maps and ACLs are shown diagrammatically in **Figure 62-11** below.

Figure 62-11: Policy-based routing example - ClassMaps and ACLs



Step 1: Create VLANs on the switch

- Create VLANs 10, 20, and 30
- Apply these VLANs to their appropriate local ports

Practical example: Create VLAN `10` and apply it to `port1.0.2-port1.0.8`.

```
awplus# configure terminal
awplus(config)# interface port port1.0.2-port1.0.8
awplus(config)# switchport mode access
awplus(config)# switchport access vlan 10
```

- Create VLANs 110, 120, and 130
- Apply these VLANs to their appropriate WAN ports

Practical example: Create VLAN `110` and apply it to `port1.0.1`.

```
awplus# configure terminal
awplus(config)# interface port port1.0.1
awplus(config)# switchport mode access
awplus(config)# switchport access vlan 110
```

Step 2: Create access control lists (ACLs) that will match the data flows between local user devices.

- access-list 3000 permit ip 10.10.0.0/16 10.20.0.0/16
  Matches for packets from the Health user devices to Welfare user devices.
- access-list 3001 permit ip 10.10.0.0/16 10.30.0.0/16
  Matches packets from the Health user devices to Employment user devices.
- access-list 3002 permit ip 10.20.0.0/16 10.10.0.0/16
  Matches packets from the Welfare user devices to Health user devices.
- access-list 3003 permit ip 10.20.0.0/16 10.30.0.0/16
  Matches packets from the Welfare user devices to Employment user devices.
- access-list 3004 permit ip 10.30.0.0/16 10.10.0.0/16
  Matches packets from the Employment user devices to Health user devices.
- access-list 3005 permit ip 10.30.0.0/16 10.20.0.0/16
  Matches packets from the Employment user devices to Welfare user devices.

Practical example: Create an ACL that matches packets from the Health user devices to Welfare user devices.

```
awplus# configure terminal
awplus(config)# access-list 3000 permit ip 10.10.0.0/16
10.20.0.0/16
```

Step 3: Create access control lists (ACLs) that will match the data flows between user devices and all other destinations.

- access-list 3006 permit ip 10.10.0.0/16 any
  Matches packets from Health user devices to all other destinations.
- access-list 3007 permit ip 10.20.0.0/16 any
  Matches packets from Welfare user devices all other destinations.
- access-list 3008 permit ip 10.30.0.0/16 any
  Matches packets from Employment user devices to all other destinations.

Practical example: Matches packets from the Health user devices to all other destinations.

```
awplus# configure terminal
awplus(config)# access-list 3006 permit ip 10.10.0.0/16 any
```

Step 4: Setup class-maps for each department and apply an access-list to each of the class-maps.

- class-map CM-Health-to-Welfare
  Creates a class-map called **CM-Health-to-Welfare**
- match access-group 3000
  Applies the access-list 3000 to the **CM-Health-to-Welfare** class-map, so that this class-map applies to all packets matching this ACL.
- class-map CM-Health-to-Employment

Creates a class-map called **CM-Health-to-Employment.**

- match access-group 3001

  Applies the access-list 3001 to the **CM-Health-to-Employment** class-map, so that this class-map applies to all packets matching this ACL.

- class-map CM-Welfare-to-Health

  Creates a class-map called **CM-Welfare-to-Health.**

- match access-group 3002

  Applies the access-list 3002 to the **CM-Welfare-to-Health** class-map, so that this class-map applies to all packets matching this ACL.

- class-map CM-Welfare-to-Employment

  Creates a class-map called **CM-Welfare-to-Employment.**

- match access-group 3003

  Applies the access-list 3003 to the **CM-Welfare-to-Employment** class-map, so that this class-map applies to all packets matching this ACL.

- class-map CM-Employment-to-Health

  Creates a class-map called Employment-to-Health.

- match access-group 3004

  Applies the access-list 3004 to the **CM-Employment-to-Health** class-map, so that this class-map applies to all packets matching this ACL.

- class-map CM-Employment-to-Health

  Creates a class-map called **CM-Employment-to-Health.**

- match access-group 3005

  Applies the access-list 3005 to the class-map **CM-Employment-to-Health** class-map, so that this class-map applies to all packets matching this ACL.

Practical example: Create the class-map CM-Health-to-Welfare, then apply access list `3000` to it.

```
awplus# configure terminal
awplus(config)# class-map CM-Health-to-Welfare
awplus(config-cmap)# match access-group 3000
```

Step 5: Setup class-maps for each department's wide area connection and apply an access-list to each of these class-maps.

- class-map CM-Health-to-WAN

  Creates a class-map called CM-Health-to-WAN.

- match access-group 3006

  Applies the access-list 3006 to the **CM-Health-to-WAN** class-map, so that this class-map applies to all packets matching this ACL.

- class-map CM-Welfare-to-WAN

  Creates a class-map called Welfare-to-WAN

- match access-group 3007

  Applies the access-list 3007 to the **CM-Welfare-to-WAN** class-map, so that this class-map applies to all packets matching this ACL.

- class-map CM-Employment-to-WAN

Creates a class-map called Employment-to-WAN.

- match access-group 3008

  Applies the access-list 3008 to the **CM-Employment-to-WAN** class-map, so that this class-map applies to all packets matching this ACL.

Practical example: Create the class-map Health-to-WAN, then apply access list `3006` to it.

```
awplus# configure terminal
awplus(config)# class-map CM-Health-to-WAN
awplus(config-cmap)# match access-group 3006
```

Step 6: Create the Departmental Policy-Maps and associate them with their appropriate Class-Maps.

- policy-map PM-Health

  Creates the policy-map called PM-Health

- class CM-Health-to-Welfare
- class CM-Health-to-Employment

  Attaches the local Health class-maps to the PM-Health policy-map. Note that no action is applied to these two class-maps. Packets that match either of these two class-maps will be forwarded across the local network using normal routing / forwarding procedures.

- class CM-Health-to-WAN
- set ip next-hop 172.34.7.5

  Attaches the CM-Health-to-WAN class-map to this policy-map, and gives it a policy-routing action.

- policy-map PM-Welfare

  Creates the policy-map called PM-Welfare

- class CM-Welfare-to-Health
- class CM-Welfare-to-Employment

  Attaches the local Welfare class-maps to the PM-Welfare policy-map. Note that no action is applied to these two class-maps. Packets that match either of these two class-maps will be forwarded across the local network using normal routing / forwarding procedures.

- class CM-Welfare-to-WAN
- set ip next-hop 18.25.2.6

  Attaches the Welfare-to-WAN class-map to this policy-map, and gives it a policy-routing action.

- policy-map PM-Employment

  Creates the policy-map called PM- Employment

- class CM-Employment-to-Health
- class CM-Employment-to-Welfare

  Attaches the local Employment class-maps to the PM-Employment policy-map. Note that no action is applied to these two class-maps. Packets that match either of these

two class-maps will be forwarded across the local network using normal routing / forwarding procedures.

- class CM-Employment-to-WAN
- set ip next-hop 34.56.4.5

  Attaches the Employment-to-WAN class-map to this policy-map, and gives it a policy-routing action.

Practical example: Create the policy-map called PM-Employment and attach its appropriate classmaps.

```
awplus# configure terminal
awplus(config)# policy-map PM-Employment
awplus(config-pmap)# class CM-Employment-to-Health
awplus(config-pmap-c)# exit
awplus(config-pmap)# class CM-Employment-to-Welfare
awplus(config-pmap-c)# exit
awplus(config-pmap)# class CM-Employment-to-WAN
awplus(config-pmap-c)# set ip next-hop 34.56.4.5
```

Step 7: Apply these Policy Maps to appropriate ports.

- service-policy input Health-to-WAN
- service-policy input Welfare-to-WAN
- service-policy input Employment-to-WAN

Practical example: To apply a policy-map named PM-Employment to `port 1.0.18-port1.0.24`:

```
awplus# configure terminal
awplus(config)# interface port1.0.18-port1.0.24
awplus(config-if)# service-policy input PM-Employment
```

Chapter 63: QoS Commands

Command List 63.2
class 63.2
class-map 63.3
clear mls qos interface policer-counters 63.4
default-action 63.5
description (QoS policy-map) 63.6
egress-rate-limit 63.7
match access-group 63.8
match cos 63.9
match dscp 63.10
match eth-format protocol 63.11
match inner-cos 63.14
match inner-vlan 63.15
match ip-precedence 63.16
match mac-type 63.17
match tcp-flags 63.18
match vlan 63.19
mls qos cos 63.20
mls qos enable 63.21
mls qos map cos-queue to 63.22
mls qos map premark-dscp to 63.23
no police 63.25
police single-rate action 63.26
police twin-rate action 63.28
policy-map 63.30
priority-queue 63.31
remark-map 63.32
remark new-cos 63.34
service-policy input 63.36
set ip next-hop (PBR) 63.37
show class-map 63.38
show mls qos 63.39
show mls qos interface 63.40
show mls qos interface policer-counters 63.42
show mls qos interface queue-counters 63.43
show mls qos interface storm-status 63.44
show mls qos maps cos-queue 63.45
show mls qos maps premark-dscp 63.46
show policy-map 63.47
storm-action 63.48
storm-downtime 63.49
storm-protection 63.50
storm-rate 63.51
storm-window 63.52
trust dscp 63.53
wrr-queue disable queues 63.54
wrr-queue egress-rate-limit queues 63.55
wrr-queue weight queues 63.56

Command List

This chapter provides an alphabetical reference for Quality of Service commands. For more information, see **Chapter 62, Quality of Service (QoS) Introduction** and **Chapter 57, Access Control Lists Introduction**.

class

Use this command to associate an existing class-map to a policy or policy-map (traffic classification), and to enter Policy Map Class Configuration mode to configure the class-map.

Use the **no** variant of this command to delete an existing class-map.

For more information on class-maps and policy-maps, see the following sections:**"Class Maps" on page 62.7** and **"Policy Maps" on page 62.10**.

Note that if your class-map does not exist, you can create it by using the **class-map** command.

Syntax

```
class {<name>|default}
no class <name>
```

Parameter	Description
<name>	Name of the (already existing) class-map.
`default`	Specify the default class-map.

Mode Policy Map Configuration

Example The following example creates the policy-map `pmap1`(using the policy-map command), then associates this to an already existing class-map named `cmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)#
```

Related Commands **class-map**
policy-map

class-map

Use this command to create a class-map.

Use the **no** variant of this command to delete the named class-map.

Syntax

```
class-map <name>
no class-map <name>
```

Parameter	Description
<name>	Name of the class-map to be created.

Mode Global Configuration

Example This example creates a class-map called `cmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)#
```

clear mls qos interface policer-counters

Resets an interface's policer counters to zero. This can either be for a specific class-map or for all class-maps.

Syntax

```
clear mls qos interface <port> policer-counters
   [class-map <class-map>]
```

Parameter	Description
<port>	The port may be a switch port (e.g. `port1.0.4`), a static channel group (e.g. `sa3`), or a dynamic (LACP) channel group (e.g. `po4`).
`class-map`	Select a class-map.
<class-map>	Class-map name.

Mode Privileged Exec

Example To reset the policy counters to zero for all class-maps for `port1.0.1`, use the command:

```
awplus# clear mls qos interface port1.0.1 policer-counters
```

Related Commands show mls qos interface policer-counters

default-action

Sets the action for the default class-map belonging to a particular policy-map. The action for a non-default class-map depends on the action of any ACL that is applied to the policy-map.

The default action can therefore be thought of as specifying the action that will be applied to any data that does not meet the criteria specified by the applied matching commands.

Use the **no** variant of this command to reset to the default action of 'permit'.

Syntax

```
default-action [permit|deny|send-to-cpu|copy-to-cpu|copy-to-mirror|
   send-to-mirror]

no default-action
```

Parameter	Description
`permit`	Packets to permit.
`deny`	Packets to deny.
`send-to-cpu`	Specify packets to send to the CPU.
`copy-to-cpu`	Specify packets to copy to the CPU.
`copy-to-mirror`	Specify packets to copy to the mirror port.
`send-to-mirror`	Specify packets to send to the mirror port.

Default The default is '`permit`'.

Mode Policy Map Configuration

Examples To set the action for the default class-map to `deny`, use the command:

```
awplus(config-pmap)# default-action deny
```

To set the action for the default class-map to `copy-to-mirror` for use with the **mirror interface** command, use the command:

```
awplus(config-pmap)# default-action copy-to-mirror
```

Related Commands **mirror interface**

description (QoS policy-map)

Adds a textual description of the policy-map. This can be up to 80 characters long.

Use the **no** variant of this command to remove the current description from the policy-map.

Syntax

```
description <line>
no description
```

Parameter	Description
<line>	Up to 80 character long line description.

Mode Policy Map Configuration

Example To add the description, VOIP traffic, use the commands:

```
awplus(config-pmap)# description VOIP traffic
```

egress-rate-limit

Sets a limit on the amount of traffic that can be transmitted per second from this port.

Use the **no** variant of this command to disable the limiting of traffic egressing on the interface.

Syntax

```
egress-rate-limit <bandwidth>
no egress-rate-limit
```

Parameter	Description
<bandwidth>	Bandwidth <1-10000000 units per second> (usable units: k, m, g). The egress rate limit can be configured in multiples of 64kbps. If you configure a value that is not an exact multiple of 64kbps, then the value will be rounded up to the nearest higher exact multiple of 64kbps. The minimum is 64 Kb. The default unit is Kb (**k**), but Mb (**m**) or Gb (**g**) can also be specified. The command syntax is not case sensitive, so a value such as 20m or 20M will be taken to mean 20 megabits.

Mode Interface Configuration

Examples To enable egress rate limiting on a port, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# egress-rate-limit 64k
% Egress rate limit has been set to 64 Kb
```

To disable egress rate limiting on a port, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no egress-rate-limit
```

match access-group

Use this command to define match criterion for a class-map.

Syntax

```
match access-group {<hw-IP-ACL>|<hw-MAC-ACL>|<hw-named-ACL>}
no match access-group {<hw-IP-ACL>|<hw-MAC-ACL>|<hw-named-ACL>}
```

Parameter	Description
<hw-IP-ACL>	Specify a hardware IP ACL number in the range <3000-3699>.
<hw-MAC-ACL>	Specify a hardware MAC ACL number in the range <4000-4699>.
<hw-named-ACL>	Specify the hardware named ACL.

Mode Class Map Configuration

Usage First create an access-list that applies the appropriate permit/deny requirements. Then use the **match access-group** command to apply this access-list for matching to a class-map. Note that this command will apply the access-list matching only to *incoming* data packets.

Examples To configure a class-map named `cmap1` with one match criterion: `access-list 3001`, which allows IP traffic from any source to any destination, use the commands:

```
awplus# configure terminal
awplus(config)# access-list 3001 permit ip any any
awplus(config)# class-map cmap1
awplus(config-cmap)# match access-group 3001
```

To configure a class-map named `cmap2` with one match criterion: `access-list 3001`, which allows MAC traffic from any source to any destination, use the commands:

```
awplus# configure terminal
awplus(config)# access-list 4001 permit any any
awplus(config)# class-map cmap2
awplus(config-cmap)# match access-group 4001
```

To configure a class-map named `cmap3` with one match criterion: `access-list hw_acl`, which allows IP traffic from any source to any destination, use the commands:

```
awplus# configure terminal
awplus(config)# access-list hardware hw_acl
awplus(config-ip-hw-acl)# permit ip any any
awplus(config)# class-map cmap3
awplus(config-cmap)# match access-group hw_acl
```

Related Commands class-map

match cos

Sets the CoS for a class-map to match on.

Use the **no** variant of this command to remove CoS.

Syntax

```
match cos <0-7>
no match cos
```

Parameter	Description
<0-7>	Specify the CoS value.

Mode Class Map Configuration

Examples To set the class-map's CoS to `4`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match cos 4
```

To remove CoS from a class-map, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match cos
```

match dscp

Use this command to define the DSCP to match against incoming packets.

Use the **no** variant of this command to remove a previously defined DSCP.

Syntax

```
match dscp <0-63>
no match dscp
```

Parameter	Description
<0-63>	Specify DSCP value (only one value can be selected).

Mode Class Map Configuration

Usage Use the **match dscp** command to define the match criterion after creating a class-map.

Examples To configure a class-map named `cmap1` with criterion that matches IP DSCP 56, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match dscp 56
```

To remove a previously defined DSCP from a class-map named cmap1, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match dscp
```

Related Commands class-map

match eth-format protocol

This command sets the Ethernet format and the protocol for a class-map to match on.

Select one Layer 2 format and one Layer 3 protocol when you issue this command.

Use the **no** variant of this command to remove the configured Ethernet format and protocol from a class-map.

Syntax `match eth-format <layer-two-format> protocol <layer-three-protocol>`

`no match eth-format protocol`

Parameter	Description
<layer-two-formats>	
`802dot2-tagged`	802.2 Tagged Packets (enter the parameter name).
`802dot2-untagged`	802.2 Untagged Packets (enter the parameter name).
`ethii-tagged`	EthII Tagged Packets (enter the parameter name).
`ethii-untagged`	EthII Untagged Packets (enter the parameter name).
`netwareraw-tagged`	Netware Raw Tagged Packets (enter the parameter name).
`netwareraw-untagged`	Netware Raw Untagged Packets (enter the parameter name).
`snap-tagged`	SNAP Tagged Packets (enter the parameter name).
`snap-untagged`	SNAP Untagged Packets (enter the parameter name).
<layer-three-protocols>	
`<word>`	A Valid Protocol Number in hexidecimal.
`any`	Note that the parameter “any” is only valid when used with the netwarerawtagged and netwarerawuntagged protocol options.
`sna-path-control`	Protocol Number 04 (enter the parameter name or its number).
`proway-lan`	Protocol Number 0E (enter the parameter name or its number).
`eia-rs Protocol`	Number 4E (enter the parameter name or its number).
`proway Protocol`	Number 8E (enter the parameter name or its number).
`ipx-802dot2`	Protocol Number E0 (enter the parameter name or its number).
`netbeui`	Protocol Number F0 (enter the parameter name or its number).
`iso-clns-is`	Protocol Number FE (enter the parameter name or its number).

Parameter(cont.)	Description(cont.)
xdot75-internet	Protocol Number 0801(enter the parameter name or its number).
nbs-internet	Protocol Number 0802 (enter the parameter name or its number).
ecma-internet	Protocol Number 0803 (enter the parameter name or its number).
chaosnet	Protocol Number 0804 (enter the parameter name or its number).
xdot25-level-3	Protocol Number 0805 (enter the parameter name or its number).
arp Protocol	Number 0806 (enter the parameter name or its number).
xns-compat	Protocol Number 0807 (enter the parameter name or its number).
banyan-systems	Protocol Number 0BAD (enter the parameter name or its number).
bbn-simnet	Protocol Number 5208 (enter the parameter name or its number).
dec-mop-dump-ld	Protocol Number 6001 (enter the parameter name or its number).
dec-mop-rem-cdons	Protocol Number 6002 (enter the parameter name or its number).
dec-decnet	Protocol Number 6003 (enter the parameter name or its number).
dec-lat	Protocol Number 6004 (enter the parameter name or its number).
dec-diagnostic	Protocol Number 6005 (enter the parameter name or its number).
dec-customer	Protocol Number 6006 (enter the parameter name or its number).
dec-lavc	Protocol Number 6007 (enter the parameter name or its number).
rarp	Protocol Number 8035 (enter the parameter name or its number).
dec-lanbridge	Protocol Number 8038 (enter the parameter name or its number).
dec-encryption	Protocol Number 803D (enter the parameter name or its number).
appletalk	Protocol Number 809B (enter the parameter name or its number).

Parameter(cont.)	Description(cont.)
ibm-sna	Protocol Number 80D5 (enter the parameter name or its number).
appletalk-aarp	Protocol Number 80F3 (enter the parameter name or its number).
snmp	Protocol Number 814Cv.
ethertalk-2	Protocol Number 809B (enter the parameter name or its number).
ethertalk-2-aarp	Protocol Number 80F3 (enter the parameter name or its number).
ipx-snap	Protocol Number 8137 (enter the parameter name or its number).
ipx-802dot3	Protocol Number FFFF (enter the parameter name or its number).
ip	Protocol Number 0800 (enter the parameter name or its number).
ipx	Protocol Number 8137 (enter the parameter name or its number).
ipv6	Protocol Number 86DD (enter the parameter name or its number).

Mode Class Map Configuration

Examples To remove the eth-format and the protocol from the class-map `cmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match eth-format protocol
```

To set the eth-format to `ethii-tagged` and the protocol to `0800` (IP) for class-map `cmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match eth-format ethii-tagged protocol
0800
```

or

```
awplus(config-cmap)# match eth-format ethii-tagged protocol ip
```

match inner-cos

Sets the Inner CoS for a class-map to match on.

Use the **no** variant of this command to remove CoS.

Syntax

```
match inner-cos <0-7>
no match inner-cos
```

Parameter	Description
<0-7>	Specify the Inner CoS value.

Mode Class Map Configuration

Examples To set the class-map's inner-cos to 4, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match inner-cos 4
```

To remove CoS from the class-map, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match inner-cos
```

match inner-vlan

Use this command to define the inner VLAN ID used as match criteria to classify a traffic class.

Use the **no** variant of this command to disable the VLAN ID used as match criteria.

Syntax

```
match inner-vlan <1-4094>
no match inner-vlan
```

Parameter	Description
<1-4094>	The VLAN number.

Mode Class Map Configuration

Usage This command is used in double-tagged networks to match on a VLAN ID belonging to the client network. For more information on VLAN double-tagged networks, see **"VLAN Double Tagging (VLAN Stacking)" on page 18.5**.

Examples To configure a class-map named `cmap1` to include traffic from inner VLAN 3, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match inner-vlan 3
```

To disable the configured VLAN ID as a match criteria for the class-map named `cmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match inner-vlan
```

match ip-precedence

Use this command to identify IP precedence values as match criteria.

Use the **no** variant of this command to remove IP precedence values from a class-map.

Syntax

```
match ip-precedence <0-7>
no match ip-precedence
```

Parameter	Description
<0-7>	The precedence value to be matched.

Mode Class Map Configuration

Example To configure a class-map named `cmap1` to evaluate all IPv4 packets for a precedence value of `5`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match ip-precedence 5
```

match mac-type

Use this command to set the MAC type for a class-map to match on.

Use **no** variant of this command to remove the MAC type match entry.

Syntax

```
match mac-type {l2bcast|l2mcast|l2ucast}
no match mac-type
```

Parameter	Description
l2bcast	Layer 2 Broadcast traffic.
l2mcast	Layer 2 Multicast traffic.
l2ucast	Layer 2 Unicast traffic.

Mode Class Map Configuration

Examples To set the class-map's MAC type to Layer 2 multicast, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match mac-type l2mcast
```

To remove the class-map's MAC type entry, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match mac-type
```

match tcp-flags

Sets one or more TCP flags (control bits) for a class-map to match on.

Use the **no** variant of this command to remove one or more TCP flags for a class-map to match on.

Syntax

```
match tcp-flags {[ack][fin][rst][syn][urg]}
no match tcp-flags {[ack][fin][rst][syn][urg]}
```

Parameter	Description
ack	Acknowledge.
fin	Finish.
rst	Reset.
syn	Synchronize.
urg	Urgent.

Mode Class Map Configuration

Examples To set the class-map's TCP flags to `ack` and `syn`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map
awplus(config-cmap)# match tcp-flags ack syn
```

To remove the TCP flags `ack` and `rst`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map
awplus(config-cmap)# no match tcp-flags ack rst
```

match vlan

Use this command to define the VLAN ID used as match criteria to classify a traffic class.

Use the **no** variant of this command to disable the VLAN ID used as match criteria.

Syntax

```
match vlan <1-4094>
no match vlan
```

Parameter	Description
<1-4094>	The VLAN number.

Mode Class Map Configuration

Examples To configure a class-map named `cmap1` to include traffic from VLAN 3, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# match vlan 3
```

To disable the configured VLAN ID as a match criteria for the class-map named `cmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# class-map cmap1
awplus(config-cmap)# no match vlan
```

mls qos cos

This command assigns a CoS (Class of Service) user-priority value to untagged frames entering a specified interface. By default, all untagged frames are assigned a CoS value of 0.

Use the **no** variant of this command to return the interface to the default CoS setting for untagged frames entering the interface.

Syntax

```
mls qos cos <0-7>
no mls qos cos
```

Parameter	Description
<0-7>	The Class of Service, user-priority value.

Default By default, all untagged frames are assigned a CoS value of 0. Note that for tagged frames, the default behavior is not to alter the CoS value.

Mode Interface Configuration

Example To assign a CoS user priority value of 2 to all untagged packets entering ports 1.0.1 to 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.6
awplus(config-if)# mls qos cos 2
```

mls qos enable

Use this command to globally enable QoS on the switch or stack.

Use the **no** variant of this command to globally disable QoS and remove all QoS configuration. The **no** variant of this command removes all class-maps, policy-maps, policers, and queue-sets that have been created. Running the **no mls qos** command will therefore remove all pre-existing QoS configurations on the switch.

Mode Global Configuration

Syntax

```
mls qos enable
no mls qos
```

Example To enable QoS on the switch, use the commands:

```
awplus# configure terminal
awplus(config)# mls qos enable
```

mls qos map cos-queue to

Use this command to set the default CoS to egress queue mapping. This is the default queue mapping for packets that do not get assigned an egress queue via any other QoS functionality.

Use the **no** variant of this command to reset the cos-queue map back to its default setting. The default mappings for this command are:

```
CoS Priority :      0 1 2 3 4 5 6 7
--------------------------------
CoS QUEUE:          2 0 1 3 4 5 6 7
```

For more information see **"Mapping CoS tags to traffic types" on page 62.12**.

Syntax

```
mls qos map cos-queue <cos-priority> to <queue-number>
no mls qos map cos-queue
```

Parameter	Description
<cos-priority>	CoS priority value. Can take a value 0 to 7.
<queue-number>	Queue number. Can take a value 0 to 7.

Mode Global Configuration

Examples To set the cos-queue map back to its defaults, use the command:

```
awplus# configure terminal
awplus(config)# no mls qos map cos-queue
```

To map CoS 2 to queue 0, use the command:

```
awplus# configure terminal
awplus(config)# mls qos map cos-queue 2 to 0
```

Related Commands **show mls qos interface**

mls qos map premark-dscp to

This command configures the premark-dscp map. It is used when traffic is classified by a class-map that has **trust dscp** configured. Based on a lookup DSCP, the map determines a new DSCP, CoS, queue and bandwidth class for the traffic.

The **no** variant of this command resets the premark-dscp map to its defaults. If no DSCP is specified then all DSCP entries will be reset to their defaults.

Syntax

```
mls qos map premark-dscp <0-63> to {[new-dscp <0-63>][new-cos <0-7>]
    [new-bandwidth-class {green|yellow|red}]}

no mls qos map premark-dscp [<0-63>]
```

Parameter	Description
`premark-dscp <0-63>`	The DSCP value on ingress.
`new-dscp <0-63>`	The DSCP value that the packet will have on egress. If unspecified, this value will remain the DSCP ingress value.
`new-cos <0-7>`	The CoS value that the packet will have on egress. If unspecified, this value will retain its value on ingress.
`new-bandwidth-class`	Modify Egress Bandwidth-class. If unspecified, this value will be set to green.
`green`	Egress Bandwidth-class green (marked down Bandwidth-class).
`yellow`	Egress Bandwidth-class yellow (marked down Bandwidth-class).
`red`	Egress Bandwidth-class red (marked down Bandwidth-class).

Mode Global Configuration

Usage With the **trust dscp** command set, this command (**mls qos map premark-dscp**) enables you to make the following changes:

1. remap the DSCP (leaving the other settings unchanged)
2. remap any or all of CoS, output queue, or bandwidth class values (leaving the DSCP unchanged)

Note If you attempt to remap both the DSCP and another setting, only the DSCP remap will take effect.

Example To set the entry for DSCP `1` to use a new DSCP of `2`, a new CoS of `3`, a new queue of `4` and a new bandwidth class of `yellow`, use the commands:

```
awplus# configure terminal
awplus(config)# mls qos map premark-dscp 1 to new-dscp 2
awplus(config)# mls qos map premark-dscp 2 to new-cos 3
awplus(config)# mls qos map premark-dscp 2 to new-queue 4
awplus(config)# mls qos map premark-dscp 2 to new-bandwidth-
class yellow
```

Example To set the entry for DSCP `1` to use a new DSCP of `2`, a new CoS of `3`, and a new bandwidth class of yellow, use the command:

```
awplus# configure terminal
awplus(config)# mls qos map premark-dscp 1 to new-dscp 2
new-cos 3 new-bandwidth-class yellow
```

Example To reset the entry for DSCP `1` use the command:

```
awplus# configure terminal
awplus(config)# no mls qos map premark-dscp 1
```

Related Commands **show mls qos maps premark-dscp**
trust dscp

no police

Disables any policer previously configured on the class-map.

Syntax `no police`

Mode Policy Map Class Configuration

Usage This command disables any policer previously configured on the class-map.

Example To disable policing on a class-map use the command:

```
awplus# configure terminal
awplus(config)# policy-map name
awplus(config-pmap)# class classname
awplus(config-pmap-c)# no police
```

Related Commands police single-rate action
police twin-rate action

police single-rate action

Configures a single-rate policer for a class-map.

Syntax

```
police single-rate <cir> <cbs> <ebs> action {drop-red|
   remark-transmit}
```

Parameter	Description	
<cir>	Specify the Committed Information Rate (CIR) (1-16000000 kbps).	
<cbs>	Specify the Committed Burst Size (CBS) (0-16777216 bytes).	
<ebs>	Specify a Excess Burst Size (EBS) (0-16777216 bytes).	
`action`	Specify the action if the rate is exceeded.	
	`drop-red`	Drop the red packets.
	`remark-transmit`	Modify the packets using the *remark map*, then transmit. You can configure the remark map using the **remark-map** command on page 63.32.

Mode Policy Map Class Configuration

Usage You can use a policer to meter the traffic classified by the class-map and assign it to one of three bandwidth classes.

The bandwidth classes are green (conforming), yellow (partially-conforming), and red (non-conforming). A single-rate policer is based on three values. These are the average rate, minimum burst and maximum burst.

Color	Definition
`green`	The traffic rate is less than the average rate and minimum burst.
`yellow`	The traffic rate is between the minimum burst and the maximum burst.
`red`	The traffic rate exceeds the average rate and the maximum burst.

Using an action of drop-red means that any packets classed as red are discarded.

Note This command will not take effect when applied to a class-map that attaches to a channel group whose ports span processor instances.

Example To configure a single rate meter measuring traffic of 10 Mbps that drops a sustained burst of traffic over this rate, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map name
awplus(config-pmap)# class classname
awplus(config-pmap-c)# police single-rate 10000 1875000 1875000
action drop-red
```

Related Commands
no police
police twin-rate action
remark-map

police twin-rate action

Configures a twin-rate policer for a class-map.

Syntax

```
police twin-rate <cir> <pir> <cbs> <pbs> action {drop-red|
    remark-transmit}
```

Parameter	Description	
<cir>	Specify the Committed Information Rate (CIR) (1-16000000 kbps).	
<pir>	Specify the Peak Information Rate (PIR) (kbps).	
<pbs>	Specify the Peak Burst Size (PBS) (0-16777216 bytes).	
`action`	Specify the action if rate is exceeded.	
	`drop-red`	Drop the red packets.
	`remark-transmit`	Modify the packets using the *remark map*, then transmit. You can configure the remark map using the **remark-map** command on page 63.32.

Mode Policy Map Class Configuration

Usage A policer can be used to meter the traffic classified by the class-map and as a result will be given one of three bandwidth classes. These are green (conforming), yellow (partially-conforming), and red (non-conforming).

A twin-rate policer is based on four values. These are the minimum rate, minimum burst size, maximum rate, and maximum burst size.

Bandwidth Class	Definition
`green`	The sum of the number of existing (buffered) bytes plus those arriving at the port per unit time results in a value that is less than that set for the CBS.
`yellow`	The sum of the number of existing (buffered) bytes plus those arriving at the port per unit time results in a value that is between those set for the CBS and the PBS.
`red`	The sum of the number of existing (buffered) bytes plus those arriving at the port per unit time, result in a value that exceeds that set for the PBS.

Using an action of drop-red means that any packets classed as red will be discarded.

When using an action of remark-transmit the packet will be remarked with the values configured in the policed-dscp map. The index into this map is determined by the DSCP in the packet.

Example To configure a twin rate meter measuring a minimum rate of 10 Mbps and a maximum rate of 20 Mbps that uses the premark map to remark any non-conforming traffic, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map name
awplus(config-pmap)# class classname
awplus(config-pmap-c)# police twin-rate 10000 20000 1875000
3750000 action remark-transmit
```

Related Commands **no police**
police twin-rate action

policy-map

Use this command to create a policy-map and to enter Policy Map Configuration mode to configure the specified policy-map.

Use the **no** variant of this command to delete an existing policy-map.

Syntax

```
policy-map <name>
no policy-map <name>
```

Parameter	Description
<name>	Name of the policy-map.

Mode Global Configuration

Example To create a policy-map called `pmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)#
```

Related Commands class-map

priority-queue

Configures strict priority based scheduling on the specified egress queues. You must specify at least one queue.

Syntax

```
priority-queue [0][1][2][3][4][5][6][7]
```

Parameter	Description
`[0][1]...[7]`	Specify the queues that will use strict priority scheduling. With strict priority scheduling, the switch will completely empty the highest numbered queue first, then start processing the next lowest numbered queue.

Mode

Interface Configuration.

Usage

By default, the queues on all ports are set for priority queuing. You can change the queue emptying sequence to weighted round robin, by using the **wrr-queue weight queues** command on page 63.56. You can then use the **priority-queue** command to reset the selected queues to priority queuing.

Note that the emptying sequence for priority queuing is always highest queue number to lowest queue number.

Example

To apply priority based scheduling to egress queues 1 and 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# priority-queue 1 2
```

Related Commands

show mls qos interface
show mls qos interface queue-counters
wrr-queue weight queues

remark-map

Use this command to configure the remark map. This command is applied when a policer is configured with the **action** parameter of the command **police single-rate action** set to **remark-transmit**.

The **no** variant of this command resets the remark map to its defaults. Specifying the bandwidth class is optional. If no bandwidth class is specified, then all bandwidth classes are reset to their defaults.

Syntax

```
remark-map [bandwidth-class {green|yellow|red}] to {[new-dscp <0-63>]
    [new-bandwidth-class {green|yellow|red}]}

no remark-map [bandwidth-class {green|yellow|red}] to {[new-dscp
    <0-63>] [new-bandwidth-class {green|yellow|red}]}
```

Parameter	Description
`bandwidth-class`	Specify the bandwidth class of packets to remark.
`green`	Remark green packets.
`yellow`	Remark yellow packets.
`red`	Remark red packets.
`new-dscp`	Specify the new DSCP value.
<0-63>	The DSCP value.
`new-bandwidth-class`	Specify the new bandwidth class.
`green`	Remark the packet green.
`yellow`	Remark the packet yellow.
`red`	Remark the packet red.

Mode Policy Map Class Configuration

Examples To remark the policed green traffic to a new DSCP of `2` and a new bandwidth class of `yellow`, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark-map bandwidth-class green to
new-dscp 2 new-bandwidth-class yellow
```

To remark the policed green traffic to a new DSCP of 2, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark-map bandwidth-class green to
new-dscp 2
```

To reset the DSCP for all bandwidth classes, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# no remark-map to new-dscp
```

Related Commands **police single-rate action**
police twin-rate action

remark new-cos

This command enables you to configure and remark either or both of:

- the CoS flag in the data packet
- the input into the CoS to queue map, thus changing the destination egress queue.

Syntax

```
remark new-cos <0-7> [internal|external|both]
no remark new-cos [internal|external|both]
```

Parameter	Description
<0-7>	The new value for the CoS flag and/or the input into the CoS to queue map.
external	Remarks the CoS flag in the packet.
internal	Remarks the new-CoS input into the CoS to queue map.
both	Remarks (with the same value) both the CoS flag in the packet and the input to the CoS to queue map.

Mode Policy Map Class Configuration

Usage The default CoS to Queue mappings are shown in the following table:

CoS Value	0	1	2	3	4	5	6	7
Egress Queue No	2	0	1	3	4	5	6	7

The relationship between this command and the CoS to queue map is shown in Figure 63-1.

Figure 63-1: Remarking and the CoS to Q map

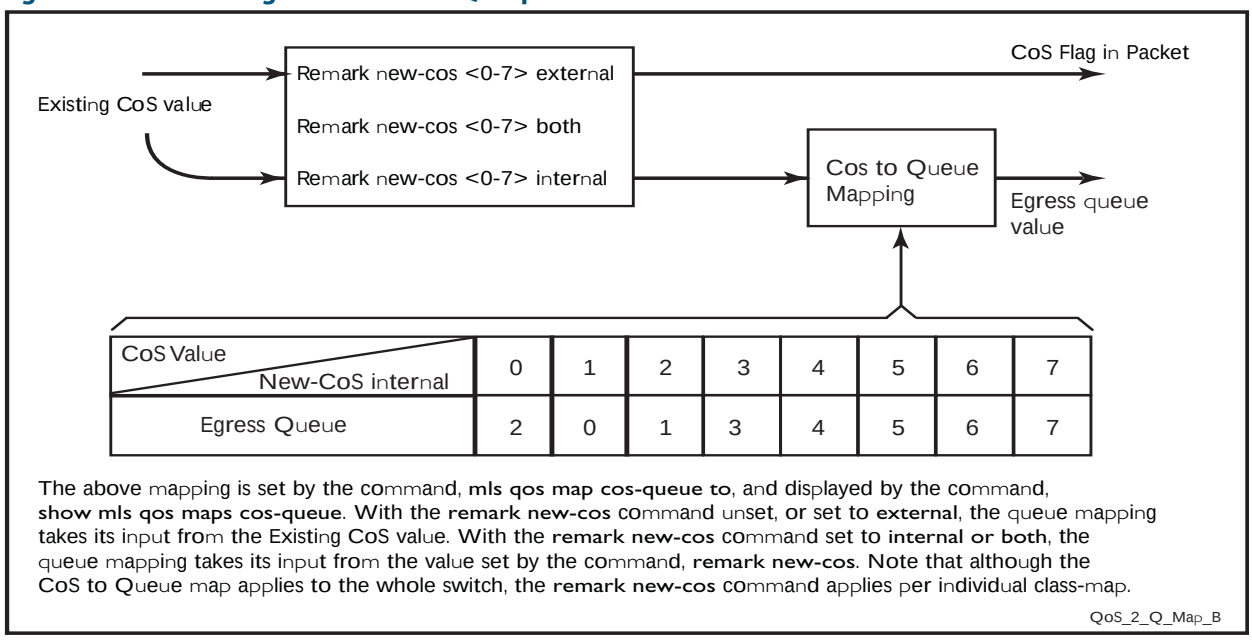


Table 63-1: CoS to egress queue remarking function

Input	Command	Output
CoS field = 1	Remark new-cos (not configured)	CoS value = 1 Packet sent to egress queue 0
CoS field = 1	Remark new-cos 2 external	CoS value = 2 Packet sent to egress queue 0
CoS set to 1	Remark new-cos 2 internal	CoS value = 1 Packet sent to egress queue 1
CoS set to 1	Remark new-cos 2 both	CoS value = 2 Packet sent to egress queue 1
Note: This table assumes that the CoS to Queue map is set to its default values.		

Example For policy-map `pmap3` and class-map `cmap1`, set the CoS value to `2` and also set the input to the CoS to queue map so that the traffic is assigned to egress queue 1:

```
awplus# configure terminal
awplus(config)# policy-map pmap3
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# remark new-cos 2 both
```

Related Commands **mls qos map cos-queue to**
show mls qos maps cos-queue

service-policy input

Use this command to apply a policy-map to the input of an interface.

Use the **no** variant of this command to remove a policy-map and interface association.

Syntax

```
service-policy input <policy-map>
no service-policy input <policy-map>
```

Parameter	Description
<policy-map>	Policy map name that the input will applied to.

Mode Interface Configuration

Usage This command can be applied to switch ports or static channel groups, but not to dynamic (LACP) channel groups.

Example To apply a policy-map named `pmap1` to interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# service-policy input pmap1
```

set ip next-hop (PBR)

Forwards traffic matching this class-map to the specified next hop.

When this command is set, all packets that match a selected class-map will be forwarded to the specified next hop.

The **no** variant of this command removes the next-hop address (in the context of its policy-map and class-map) from the configuration.

Syntax

```
set ip next-hop <ip-addr>
no set ip next-hop
```

Parameter	Description
<ip-addr>	The IP address of the next hop destination.

Mode Policy Map Class Configuration

Usage In typical deployments of policy-based routing, some traffic types require normal routing (i.e. via the routes in the IP routing table) while other traffic types require policy based routing.

Where the traffic to be policy routed is a subset of the traffic that is to be normally routed, then the configuration is reasonably simple. The policy-map will contain one or more classes that match the traffic to be policy routed, and will have their next-hop configured by this command - **set ip next-hop (PBR).** The remaining traffic will be conventionally routing routed according to the rules set for the default class - providing that this is not subject to the **set ip next-hop (PBR).**

The situation becomes a little more complex where the traffic requiring normal routing is a subset of the traffic to be policy-routed. In this situation the policy-map would need to contain one, or more, classes that match the requirement for normal routing, These classes would not be configured with a **set ip next-hop (PBR)** command. Then the remaining traffic classes that require normal routing would have the **set ip next-hop (PBR)** command applied to them. Note that this traffic could be just the default class, if ALL other traffic types were to be policy-routed.

Also note that the order in which the classes are configured in the policy-map is important; because traffic is matched against the classes in the order that they were assigned to the policy-map.

Details of a practical example of such a policy-based routing is shown in **"Policy-Based Routing" on page 62.25**.

Example To forward a packet to a 192.168.1.1, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# set ip next-hop 192.168.1.1
```

show class-map

Use this command to display the QoS class-maps to define the match criteria to classify traffic.

Syntax `show class-map <class-map name>`

Parameter	Description
<class-map name>	Name of the class-map.

Mode User Exec and Privileged Exec

Example To display the QoS class-maps to define the match criteria to classify traffic, use the command:

```
awplus# show class-map cmap1
```

Output **Figure 63-2: Example output from the show class-map command**

```
CLASS-MAP-NAME: cmap1
      Set IP DSCP: 56
      Match IP DSCP: 7
```

Related Commands class-map

show mls qos

Use this command to display whether QoS is enabled or disabled on the switch.

Syntax `show mls qos`

Mode User Exec and Privileged Exec

Example To display whether QoS is enabled or disabled, use the command:

```
awplus# show mls qos
```

Output **Figure 63-3: Example output from the show mls qos command**

```
awplus#show mls qos
 Enable
```

Related Commands mls qos enable

show mls qos interface

Displays the current settings for the interface. This includes its default CoS and queue, scheduling used for each queue, and any policies/maps that are attached.

Syntax `show mls qos interface [<port>]`

Parameter	Description
<port>	Switch port.

Mode User Exec and Privileged Exec

Example To display current CoS and queue settings for interface `port1.0.1`, use the command:

```
awplus# show mls qos interface port1.0.1
```

Output **Figure 63-4: Example output from the show mls qos interface command**

```
Default CoS:   7
    Default Queue: 7
    Number of egress queues: 8
    Queue Set: 1
    Egress Queue:         0
      Status:             Enabled
      Scheduler:          Strict Priority
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         1
      Status:             Enabled
      Scheduler:          Strict Priority
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         2
      Status:             Enabled
      Scheduler:          Strict Priority
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         3
      Status:             Enabled
      Scheduler:          Wrr Group 2
      Weight:             10
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         4
      Status:             Enabled
      Scheduler:          Wrr Group 1
      Weight:             10
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         5
      Status:             Enabled
      Scheduler:          Strict Priority
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         6
      Status:             Enabled
      Scheduler:          Strict Priority
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
    Egress Queue:         7
      Status:             Enabled
      Scheduler:          Strict Priority
      Queue Limit:        12%
      Egress Rate Limit: 0 Kb
```

Table 63-2: Parameters in the output of the show mls qos interface command

Parameter	Description
`Default CoS`	The default CoS priority that will be applied to all packets arriving on this interface.
`Default Queue`	The default queue that will be applied to all packets arriving on this interface.
`Number of egress queues`	The total number of egress queues available on this interface.
`Queue Set`	Drop queue set that has been applied to the port. This could either be operating in threshold or random-detect mode.
`Egress Queue X`	Number of this egress queue.
`Status`	Queue can either be enabled or disabled.
`Scheduler`	The scheduling mode being used for servicing the transmission of packets on this port.
`Queue Limit`	The percentage of the port's buffers that have been allocated to this queue.
`Egress Rate Limit`	The amount of traffic that can be transmitted via this queue per second. 0 Kb means there is currently no rate-limiting enabled.

show mls qos interface policer-counters

This command displays an interface's policer counters. This can either be for a specific class-map or for all class-maps attached to the interface. If no class-map is specified then all class-map policer counters attached to the interface are displayed.

Note that these counters are based on metering performed on the specified class-map. Therefore, the '`Dropped Bytes`' counter is the number of bytes dropped due to metering. This is different from packets dropped via a 'deny' action in the ACL. If a policer is configured to perform re-marking, bytes can be marked Red but are not dropped, and is shown with a value of 0 for the Dropped field and a non-0 value for the '`Red Bytes`' field.

Syntax

```
show mls qos interface <port> policer-counters
   [class-map <class-map>]
```

Parameter	Description
<port>	Switch port.
`class-map`	Select a class-map.
<class-map>	Class-map name.

Mode User Exec and Privileged Exec

Usage Note that the hardware does not record distinct counters for the number of Green or Yellow bytes, so the field marked Green/Yellow is the summation of bytes that have been marked Green or Yellow by the meter.

Example To show the counters for all class-maps attached to `port1.0.1`, use the command:

```
awplus# show mls qos interface port1.0.1 policer-counters
```

Output **Figure 63-5: Example output from show mls qos interface policer-counters**

```
awplus#show mls qos int port1.0.1 policer-counters
Interface:               port1.0.1
  Class-map:             default
    Green/Yellow Bytes:  0
    Red Bytes:           0
    Dropped Bytes:       0
    Non-dropped Bytes:   0
  Class-map:             cmap1
    Green/Yellow Bytes:  1629056
    Red Bytes:           7003200
    Dropped Bytes:       0
    Non-dropped Bytes:   8632256
```

This output shows a policer configured with remarking through '`action remark-transmit`', so although bytes are marked as Red, none are dropped. Therefore, the '`Non-dropped Bytes`' field shows a summation of Green/Yellow and Red bytes.

show mls qos interface queue-counters

This command displays an interface's egress queue counters. This can either be for a specific queue or for all queues on the interface. If no queue is specified all queue counters on the interface will be displayed.

The counters show the number of frames currently in the queue and the maximum number of frames allowed in the queue, for individual egress queues and the port's queue (which will be a sum of all egress queues).

Syntax `show mls qos interface <port> queue-counters [queue <0-7>]`

Parameter	Description
<port>	Switch port.
<0-7>	Queue.

Mode User Exec and Privileged Exec

Example To show the counters for all queues on `port1.0.1`, use the command:

```
awplus# show mls qos interface port1.0.1 queue-counters
```

Output **Figure 63-6: Example output from the show mls qos interface queue-counters command**

```
Interface port1.0.4 Queue Counters:
  Port queue length     1169
  Egress Queue length:
    Queue 0                0
    Queue 1                0
    Queue 2             1169
    Queue 3                0
    Queue 4                0
    Queue 5                0
    Queue 6                0
    Queue 7                0
```

Table 63-3: Parameters in the output of the show mls qos interface queue-counters command

Parameter	Description
`Interface`	Port we are showing the counters for.
`Port queue length`	Number of frames in the port's queue. This will be the sum of all egress queues on the port.
`Egress Queue length`	Number of frames in a specific egress queue.

show mls qos interface storm-status

Show the current configuration and status of the QoS Storm Protection (QSP) on the given port.

Syntax `show mls qos interface <port> storm-status`

Parameter	Description
<port>	Switch port.

Mode User Exec and Privileged Exec

Example To see the QSP status on `port1.0.1`, use command:

```
awplus# show mls qos interface port1.0.1 storm-status
```

Output **Figure 63-7: Example output from the show mls qos interface storm-status command**

```
Interface:           port1.0.1
Storm-Protection:    Enabled
Port-status:         Enabled
Storm Action:        vlandisable
Storm Window:        5000 ms
Storm Downtime:      0 s
Timeout Remaining:   0 s
Last read data-rate: 0 kbps
Storm Rate:          1000 kbps
```

Related Commands
storm-action
storm-downtime
storm-protection
storm-rate
storm-window

show mls qos maps cos-queue

Show the current configuration of the cos-queue map.

Syntax `show mls qos maps cos-queue`

Mode User Exec and Privileged Exec

Example To display the current configuration of the cos-queue map, use the command:

```
awplus# show mls qos maps cos-queue
```

Output **Figure 63-8: Example output from the show mls qos maps cos-queue command**

```
COS-TO-QUEUE-MAP:
      COS :           0 1 2 3 4 5 6 7
      --------------------------------
      QUEUE:          0 7 1 3 4 5 6 7
```

Related Commands mls qos map cos-queue to

show mls qos maps premark-dscp

This command displays the premark-dscp map. This map is used when the **trust dscp** command has been specified for a policy-map's class-map to replace the DSCP, CoS, queue, and bandwidth class of a packet matching the class-map based on a lookup DSCP value.

Syntax `show mls qos maps premark-dscp [<0-63>]`

Parameter	Description
<0-63>	DSCP table entry.

Mode User Exec and Privileged Exec

Example To display the premark-dscp map for DSCP `1`, use the command:

```
awplus# show mls qos maps premark-dscp 1
```

Output **Figure 63-9: Example output from the show mls qos maps premark-dscp command**

```
PREMARK-DSCP-MAP:

    DSCP 1
    Bandwidth Class        Green    Yellow   Red
    ------------------------------------------------
    New DSCP               1        -        -
    New CoS                0        -        -
    New Queue              0        -        -
    New Bandwidth Class    green    -        -
```

Related Commands **mls qos map premark-dscp to**
trust dscp

show policy-map

Displays the policy-maps configured on the switch. The output also shows whether or not they are connected to a port (attached / detached) and shows their associated class-maps.

Syntax `show policy-map [<name>]`

Parameter	Description
<name>	The name of a specific policy-map.

Mode User Exec and Privileged Exec

Example To display a listing of the policy-maps configured on the switch, use the command:

```
awplus# show policy-map
```

Output **Figure 63-10: Example output from the show policy-map command**

```
POLICY-MAP-NAME: general-traffic
  State: attached
    Default class-map action: permit
    CLASS-MAP-NAME: default
    CLASS-MAP-NAME: database-traffic
```

Related Commands service-policy input

storm-action

Sets the action to take when triggered by QoS Storm Protection (QSP). There are three available options:

- **portdisable** will disable the port in software.
- **vlandisable** will disable the port from the VLAN matched by the class-map in class-map.
- **linkdown** will physically bring the port down. The **vlandisable** requires the match vlan class-map to be present in the class-map.

The **no** variant of this command will negate the action set by the **storm-action** command.

Syntax

```
storm-action {portdisable|vlandisable|linkdown}
no storm-action
```

Parameter	Description
`portdisable`	Disable the port in software.
`vlandisable`	Disable the VLAN.
`linkdown`	Shutdown the port physically.

Mode Policy Map Class Configuration

Examples To apply the storm protection of `vlandisable` to the policy-map named `pmap2`, and the class-map named `cmap1`, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c# storm-action vlandisable
```

To negate the storm protection set on the policy-map named `pmap2`, and the class-map named `cmap1`, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c# no storm-action
```

Related Commands
storm-downtime
storm-protection
storm-rate
storm-window

storm-downtime

Sets the time to re-enable the port once disabled by QoS Storm Protection (QSP). The time is given in seconds, from a minimum of one second to maximum of 86400 seconds (i.e. one day).

The **no** variant of this command resets the time to the default value of 10 seconds.

Syntax

```
storm-downtime <1-86400>
no storm-downtime
```

Parameter	Description
<1-86400>	Seconds.

Default 10 seconds

Mode Policy Map Class Configuration

Examples To re-enable the port in 1 minute, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# storm-downtime 60
```

To re-set the port to the default (10 seconds), use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap1
awplus(config-pmap-c)# no storm-downtime
```

Related Commands

storm-action
storm-protection
storm-rate
storm-window

storm-protection

Use this command to enable the Policy Based Storm Protection (such as QSP - QoS Storm Protection). Storm protection is activated as soon as a port is enabled.

The **no** variant of this command disables Policy Based Storm Protection.

Syntax

```
storm-protection
no storm-protection
```

Default By default, storm protection is disabled.

Mode Policy Map Class Configuration

Examples To enable QSP on `cmap2` in `pmap2`, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# storm-protection
```

To disable QSP on `cmap2` in `pmap2`, use the following commands:

```
awplus# policy-map pmap2
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# no storm-protection
```

Related Commands

storm-action
storm-downtime
storm-rate
storm-window

storm-rate

Sets the data rate that triggers the storm-action. The rate is in kbps and the range is from 1kbps to 10Gbps.

Note that this setting is made in conjunction with the **storm window** command.

Use the **no** variant of this command to negate the **storm-rate** command.

Syntax

```
storm-rate <1-10000000>
no storm-rate
```

Parameter	Description
<1-10000000>	The range of the storm-rate.

Default No default

Mode Policy Map Class Configuration

Usage This setting is made in conjunction with the **storm-window** command on page 63.52.

Examples To the limit to 1Mbps, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# storm-rate 1000
```

To negate the limit set previously, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# storm-rate 1000
```

Related Commands

storm-action
storm-downtime
storm-protection
storm-window

storm-window

Sets the window size of QoS Storm Protection (QSP). This sets the time to poll the data-rate every given milliseconds. Minimum window size of 100 ms and the maximum is 60 sec.

Use the **no** variant of this command to negate the **storm-window** command.

Syntax

```
storm-window <100-60000>
no storm-window
```

Parameter	Description
<100-60000>	The window size, measured in milliseconds.

Default No default

Mode Policy Map Class Configuration

Usage This command should be set in conjunction with the **storm-rate** command on page 63.51.

Examples To set the QSP window size to `5000` ms, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# storm-window 5000
```

To negate the QSP window size set previously, use the following commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap2
awplus(config-pmap)# class cmap2
awplus(config-pmap-c)# storm-window 5000
```

Related Commands

storm-action
storm-downtime
storm-protection
storm-rate

trust dscp

This command enables the premark-dscp map to replace the bandwidth-class, CoS, DSCP, and queue of classified traffic based on a lookup DSCP value.

With the no variant of this command, no premark-dscp mapping function will be applied for the selected class-map. QoS components of the packet existing either at ingress, or applied by the class-map, will pass unchanged.

Syntax

```
trust dscp
no trust
```

Mode Policy-Map Configuration. Because policy-maps are applied to ports, you can think of **trust dscp** as a per-port setting.

Examples To enable the premark-dscp map lookup for policy-map `pmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# trust dscp
```

To disable the premark-dscp map lookup for policy-map `pmap1`, use the commands:

```
awplus# configure terminal
awplus(config)# policy-map pmap1
awplus(config-pmap)# no trust
```

Related Commands mls qos map premark-dscp to

wrr-queue disable queues

Use this command to disable an egress queue from transmitting traffic.

The **no** variant of this command enables an egress queue to transmit traffic.

Syntax

```
wrr-queue disable queues [0][1][2][3][4][5][6][7]
no wrr-queue disable queues [0][1][2][3][4][5][6][7]
```

Parameter	Description
`[0][2]...[7]`	Selects one or more queues numbered 0 to 7.

Mode Interface Configuration

Examples To enable queue 1 to transmit traffic, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no wrr-queue disable queues 1
```

To disable queue 1 from transmitting traffic, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue disable queues 1
```

Related Commands show mls qos interface

wrr-queue egress-rate-limit queues

Sets a limit on the amount of traffic that can be transmitted per second from these queues. The default unit is in Kb, but Mb or Gb can also be specified. The minimum is 651Kb.

Syntax

```
wrr-queue egress-rate-limit <bandwidth> queues
    {0}[1][2][3][4][5][6][7]

no wrr-queue egress-rate-limit <bandwidth> queues
    {0}[1][2][3][4][5][6][7]
```

Parameter	Description
<bandwidth>	Bandwidth <1-10000000 kbits> (usable units: k, m, g).
{0}[1]...[7]	Selects one or more queues to apply the bandwidth limit to as specified in the preceding *<bandwidth>* parameter.

Mode Interface Configuration

Example To limit the egress rate of queues 0, 1 and 2, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue egress-rate-limit 500M
                   queues 0 1 2
```

Related Commands show mls qos interface

wrr-queue weight queues

This command configures weighted round-robin based scheduling on the specified egress queues on switch port interfaces only. The weights are specified as ratios relative to each other. It does not matter which numbers you use, only their ratio.

Syntax `wrr-queue weight <1-15> queues [0][1][2][3][4][5][6][7]`

Parameter	Description
`<1-15>`	Weight (the higher the number the greater will be the queue servicing).
`[0][1]...[7]`	Enter egress queue numbers 0-7, to assign the specified queues the specified weight.

Mode Interface Configuration for switch port interfaces only (not for static aggregated interfaces).

Usage Only apply weighted round-robin based scheduling to switch port interfaces (for example, `awplus(config)#interface port1.0.2`).

You cannot apply weighted round-robin based scheduling to static aggregated interfaces (for example, `awplus(config)#interface sa2`). Attempting to apply weighted round-robin based scheduling on aggregated interfaces will display the console error shown below:

```
awplus# configure terminal
awplus(config)# interface sa2
awplus(config-if)# wrr-queue weight
% Invalid input detected at ^ marker
```

Example To apply a WRR weight of `6` to queues 0 and 1 on port1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# wrr-queue weight 6 queues 0 1
```

Related Commands **priority-queue**
show mls qos interface

Chapter 64: Authentication Introduction and Configuration

Authentication Introduction 64.2
Configuring a Guest VLAN 64.2
802.1X-Authentication 64.3
Web-Authentication 64.4
What is Web-Authentication? 64.4
Web-Authentication Basics 64.4
Configuring Web-Authentication 64.6
Starting a Web-Authentication Session 64.8
Configuring an External Login Page 64.10
Customizing Web-Authentication Login Page 64.13
Support for Protocols Underlying Web-Authentication 64.15
Web-Authentication Timeout Connect 64.20
Web Authorization Proxy 64.20
MAC-Authentication 64.21
Why is MAC-Authentication Required? 64.21
How Does MAC-Authentication Work? 64.21
Configuring MAC-Authentication 64.22
Tri-Authentication 64.23
Tri-Authentication Configuration 64.24
Two-Step Authentication 64.25
Ensuring Authentication Methods Require Different Usernames and Passwords 64.26
Roaming Authentication 64.27
Roaming Authentication Overview 64.28
Roaming Authentication Feature Interactions 64.29
Unauthenticated Supplicant Traffic 64.30
Deciding When a Supplicant Fails Authentication 64.32
Failed Authentication VLAN 64.33
Limitations on Allowed Feature Combinations 64.33

Authentication Introduction

Authentication commands enable you to specify three different types of device authentication: 802.1X-Authentication, Web-Authentication, and MAC-Authentication.

802.1X is an IEEE standard providing a mechanism for authenticating devices attached to a LAN port or wireless device. Web-Authentication is applicable to devices that have a human user who opens the web browser and types in a user name and password when requested. MAC-Authentication is used to authenticate devices that have neither a human user nor implement 802.1X supplicant when making a network connection request.

Configuring a Guest VLAN

In a secure network, the default behavior is to deny any access to supplicants that cannot be authenticated. However, it is often convenient to allow unauthenticated users to have limited access. A popular solution is to define a limited-access VLAN, called the Guest VLAN, and assign unauthenticated users into that VLAN. Unauthenticated supplicants are either supplicants who have attempted and failed authentication or haven't performed any authentication.

See the **auth guest-vlan** command on page 65.9 for command information about Guest VLAN.

By default, traffic from unauthenticated supplicants in the Guest VLAN will only be L2 switched within the Guest VLAN. But, if the **routing** parameter for the **auth guest vlan** command is configured, then the switch will route unauthenticated supplicants' traffic to other VLANs if required, and will relay their DHCP requests to servers in other VLANs if required.

You can configure 802.1X to accept a Dynamic VLAN assignment, or fall back to a Guest VLAN upon failure.

To configure a switch to perform 802.1X-Authentication, and assign VLAN IDs to ports where devices authentication successfully, and put non-authenticated users into a Guest VLAN, proceed as follows:

```
awplus# configure terminal
awplus(config)# radius-server host <ip-address> key
<key-string>
awplus(config)# aaa authentication dot1x default group
radius
awplus(config)# interface <interface-range>
awplus(config-if)# switchport mode access
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# auth guest-vlan 100
```

802.1X-Authentication

802.1X is an IEEE standard providing a mechanism for authenticating devices attached to a LAN port or wireless device. Devices wishing to access services behind a port must authenticate themselves before any Ethernet packets are allowed to pass through. The protocol is referred to as 802.1X because it was initially defined in the IEEE standard 802.1X, published in 2001 and revised in 2004 and again as the current 802.1X 2010 standard.

For more information about 802.1X, see **Chapter 68, 802.1X Introduction and Configuration**.

Web-Authentication

What is Web-Authentication?

Web-Authentication is a convenient alternative to 802.1X-Authentication. It's commonly used to authenticate users in educational institutions, where regular users' workstations are not managed by the network administrator. Web-Authentication enables the switch to detect an unauthenticated workstation web browsing into the network, then redirect the user's web browser to its own authentication web page.

Web-Authentication works like this:

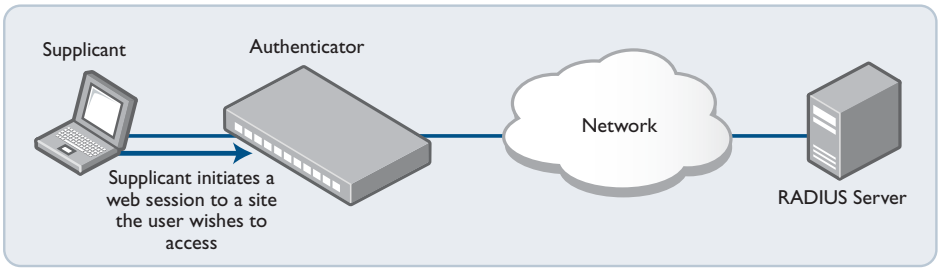
- The user enters their username and password into the web page, which the switch then sends to a RADIUS server for checking.
- If the RADIUS server accepts the user's credentials, the switch then allows their traffic into the network.

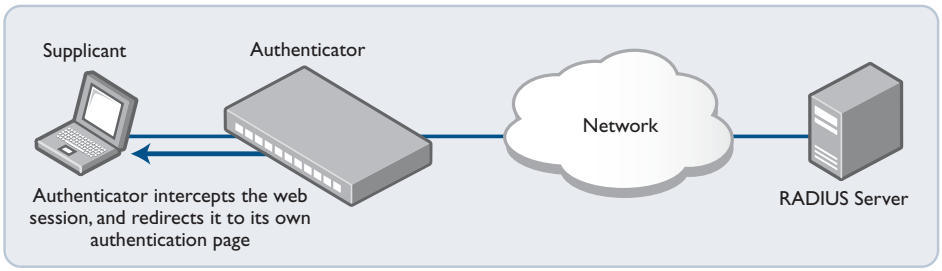
The Web-authenticating switch interacts with a RADIUS server in the same way as an 802.1X authenticator. The two methods can be used together in the same network, using the same RADIUS server.

Web-Authentication Basics

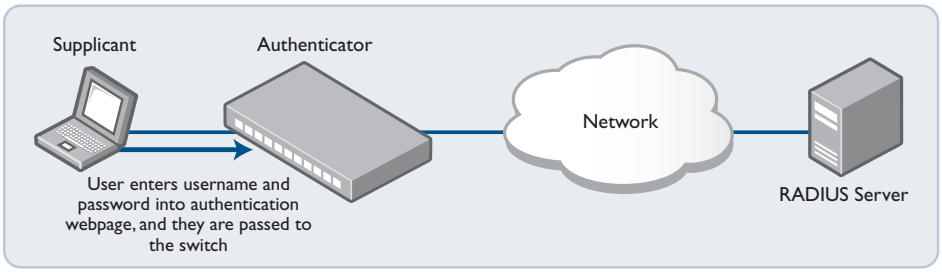
The operation of Web-Authentication is explained as below:

1. The authenticating switch receives HTTP or HTTPS traffic from an unauthenticated supplicant. It intercepts the supplicant's web session and redirects it to its own internal web server.

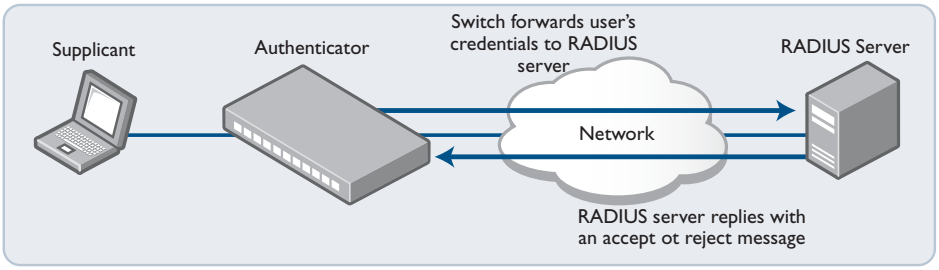




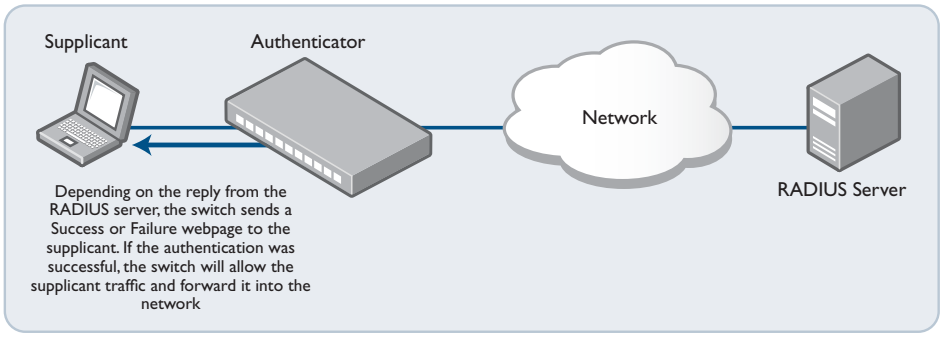
2. The web server serves up an authentication page on which the user enters their username and password.



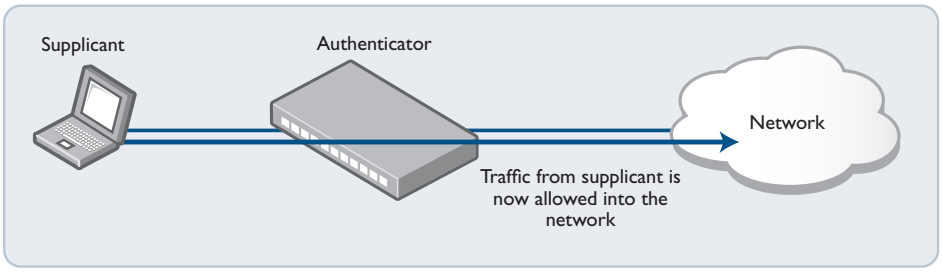
3. The username and password are sent to a RADIUS server, which informs the authenticating switch whether or not the supplicant is authenticated.



4. The user is then informed of the RADIUS server's verdict.



5. If the supplicant has been successfully authenticated, the authenticating switch will give the supplicant workstation access to the network.



Configuring Web-Authentication

Web-Authentication can be configured on a switch in the following steps:

1. Configure a RADIUS server.

```
awplus(config)# radius-server host <server-ip-address> key
<shared-secret>
```

2. Instruct Web-Authentication to use the configured RADIUS server.

```
awplus(config)# aaa authentication auth-web default group
radius
```

3. Define the IP address through which the Web-Authentication service will be accessed.

```
awplus(config)# auth-web-server ipaddress <ip-address>
```

4. Configure ports for Web-Authentication.

```
awplus# interface port1.0.1-1.0.6
awplus(config)# auth-web enable
```

Choosing the Web-Authentication server address

When you come to configure Web-Authentication, you need to answer some questions:

Questions What IP address should I specify as the Web-Authentication server address? Is it okay to use just any IP address that is configured on one of the switch's VLANs, or is the choice more constrained than that?

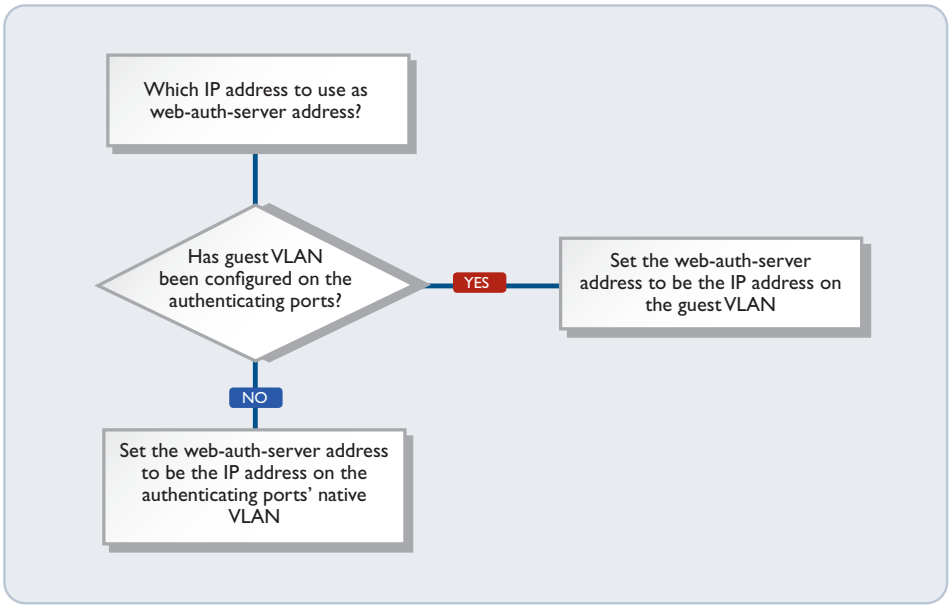
Answer You must use the IP address that is configured on the VLAN that the supplicant's packets will arrive on.

The logic that the switch uses in deciding which VLAN to associate non-authenticated supplicants' packets with is:

- If guest VLAN has been configured on the port where the packet arrives, then associate the packet with the guest VLAN.
- Otherwise associate the packet with the port's native VLAN.

If you configure the supplicant-connected ports with guest VLAN, then use the IP address on the guest VLAN as the IP address of the Web-Authentication server. Otherwise use the IP address on the supplicant-connected ports' native VLAN.

The diagram below illustrates how to decide which IP address to use as the Web-auth-server address:



Configuration Example 1: Using a guest VLAN

```
VLAN database
    VLAN 20 name guest
    VLAN 10 name edge
    VLAN 30 name core

radius-server host 192.168.30.129 key verysecret
aaa authentication auth-Web default group RADIUS
auth-Web-server ipaddress 192.168.20.1

int vlan10
    ip address 192.168.10.1/24
int vlan20
    ip address 192.168.20.1/24
int vlan30
ip address 192.168.30.1/24

int port1.0.1-1.0.3
  switchport access vlan 10
 auth-Web enable
  auth guest-vlan 20

int port1.0.4-1.0.6
  switchport access vlan 30
```

Configuration Example 2: Not using a guest VLAN

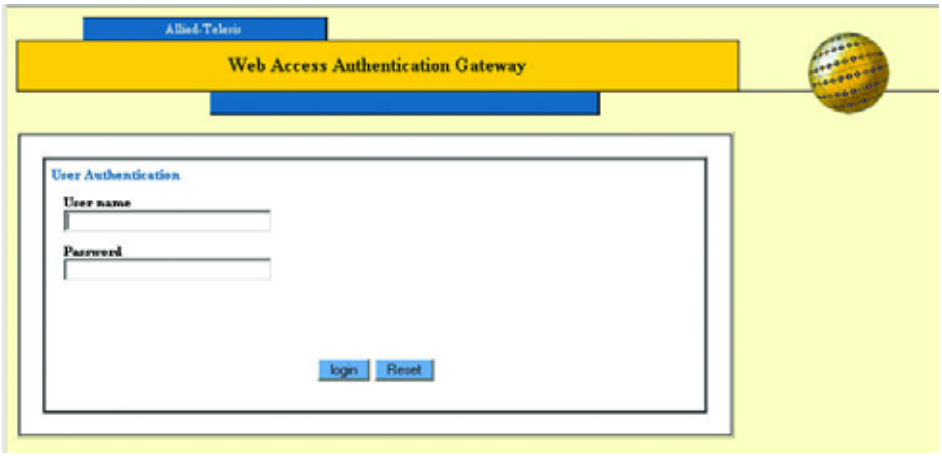
```
VLAN database
    VLAN 10 name edge
    VLAN 30 name core

radius-server host 192.168.30.129 key verysecret
aaa authentication auth-web default group radius
auth-web-server ipaddress 192.168.10.1

int vlan10
    ip address 192.168.10.1/24
int vlan30
ip address 192.168.30.1/24

int port1.0.1-1.0.3
  switchport access vlan 10
  auth-Web enable

int port1.0.4-1.0.6
  switchport access vlan 30
```

Starting a Web-Authentication Session

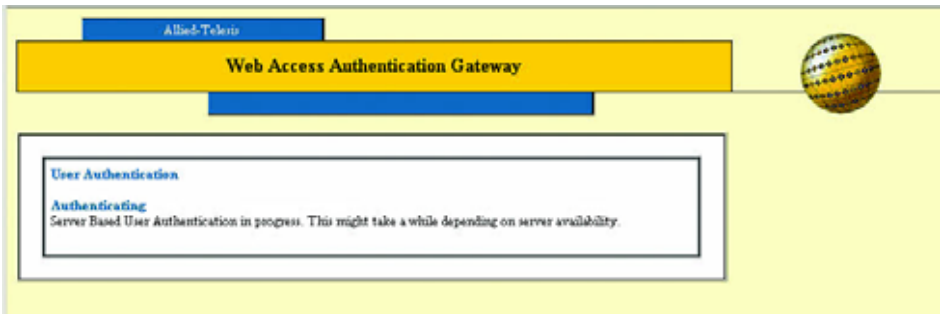
This section explains what the user actually sees in a Web-Authentication session:

1. The user starts their Web browser, and browses to a page they wish to view. Shortly thereafter, the address in the browser's address bar automatically changes to the address of the authenticating switch's authentication page.

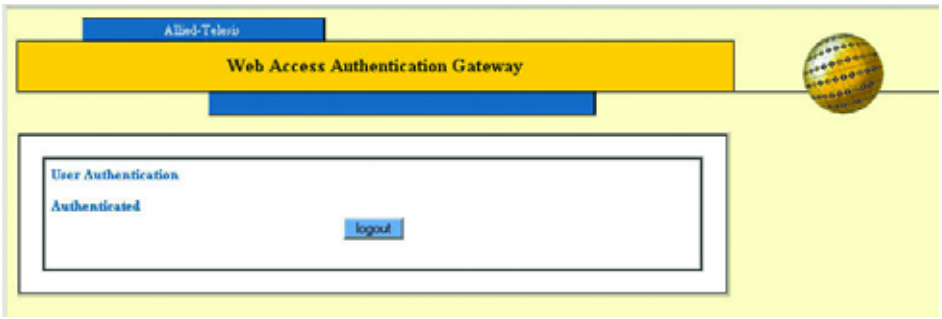
2. In the switch's authentication page, the user enters their user name and password, and clicks **login**. The maximum length of the user name and password is 64 characters. The local RADIUS server has a password length limit of 31 characters.



3. The switch displays a page that informs them that authentication is in progress.

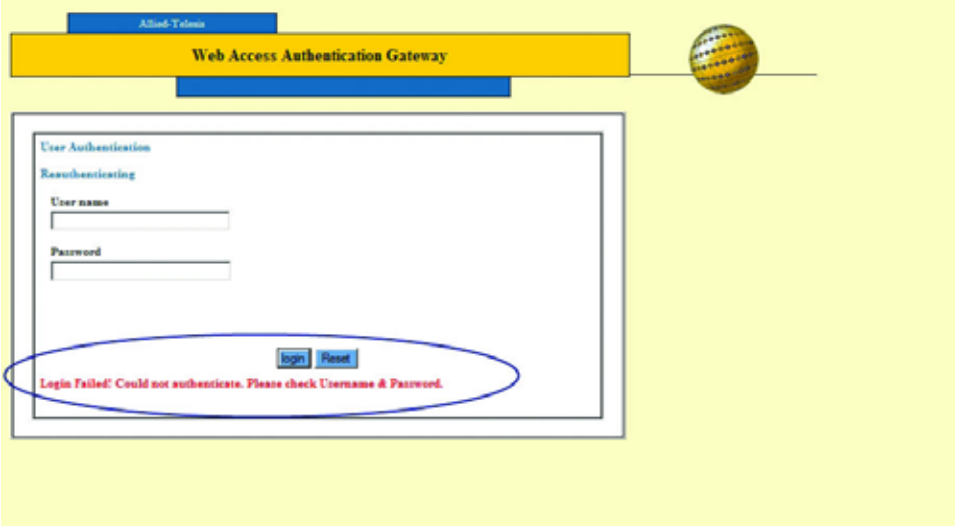


4. Once authentication is complete, the authentication result is displayed.



If the user enters a username/password combination that is not accepted by the RADIUS server, the switch presents an invitation to check the username and password.

If the user enters incorrect usernames/passwords several times, the authentication fails. The number of times a user can try to login is configurable but it is set to 10 by default.

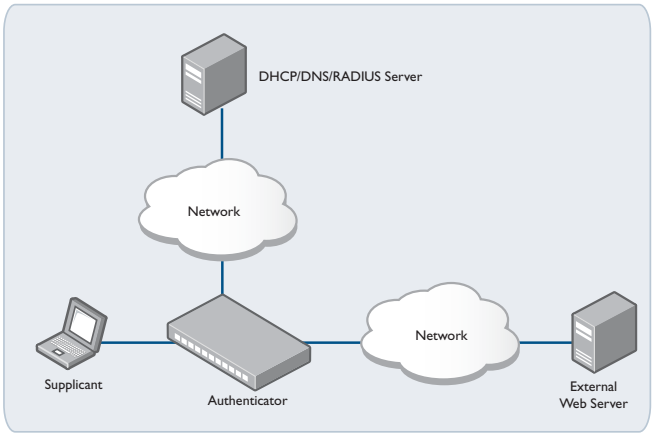


Configuring an External Login Page

This section explains how to configure an external login page for Web-Authentication rather than using the built-in AlliedWare Plus login page.

In this configuration example, a remote DHCP server is used to allocate IP addresses to the supplicants. You can also configure Authenticator as a local DHCP server.

Note that routing in the network must be set up so that packets can be routed directly between the supplicant and the external web server.



1. Configure the IP address for vlan1 to be used to reach remote DHCP/DNS/RADIUS server.

```
awplus(config)# int vlan1
awplus(config-if)# ip address 192.168.1.2/24
```

2. Configure the remote DNS server on Authenticator.

```
awplus(config)# ip name-server x.x.x.x
awplus(config)# ip dns forwarding
```

3. Configure auth-web on Authenticator

```
awplus(config)# aaa authentication auth-web default group
                radius
awplus(config)# int port1.0.1
awplus(config-if)# auth-web enable
```

4. Configure the remote RADIUS server on Authenticator. You can also configure Authenticator as a local RADIUS server.

```
awplus(config)# radius-server host 192.168.1.254 key 123
```

5. Configure the external login page by using the **auth-web-server login-url** command. You can use the domain name or the IP address for the external web server. In this example, the IP address of the external web server is 192.168.1.1. You also need to allow HTTP packets to the external login server to be forwarded by using the **auth-web forward** command. If the external web server is using another TCP port other than 80, you need to change 80 to the corresponding port number.

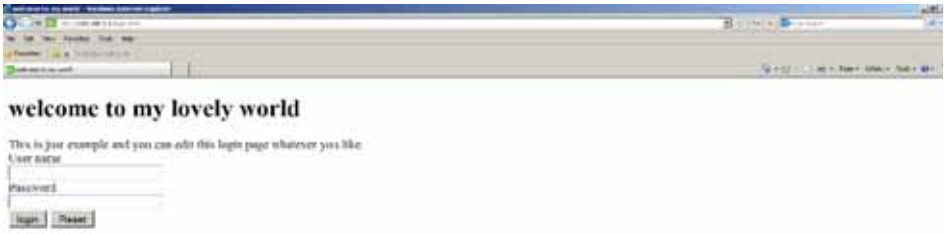
```
awplus(config)# auth-web-server login-url http://
                192.168.1.1/login.html
awplus(config-if)# auth-web forward 192.168.1.1 tcp 80
```

Supplicant Login

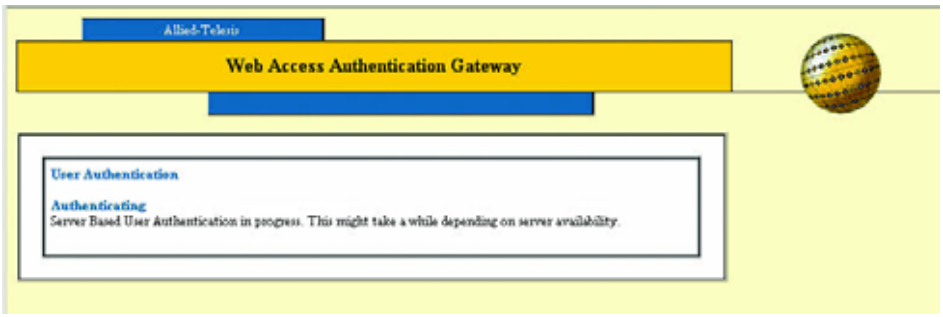
This section explains what the user actually sees in an external Web-Authentication session.

1. When a supplicant tries to access any website, the supplicant will be redirected to the external login page for authentication.

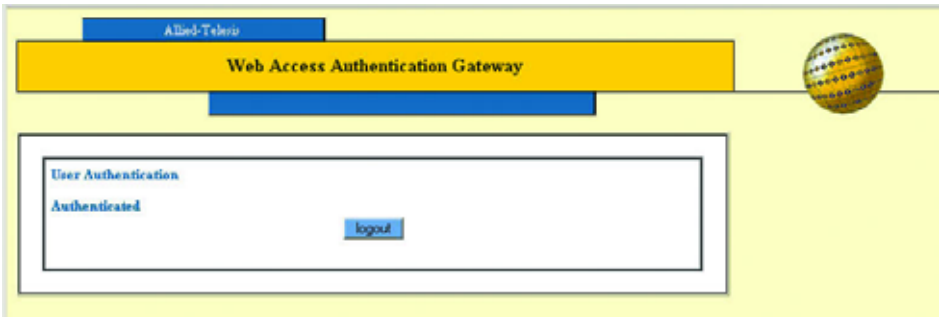
2. On the device's authentication page, the user enters their user name and password, and clicks **login**.



3. The device displays a page that informs them that authentication is in progress.



4. Once authentication is complete, the authentication result is displayed.



If the user enters a username/password combination that is not accepted by the RADIUS server, the device presents an invitation to check the username and password.

External Supplicant Login Page Support

A file named login.html must be present on the external web server. The file must contain the following elements:

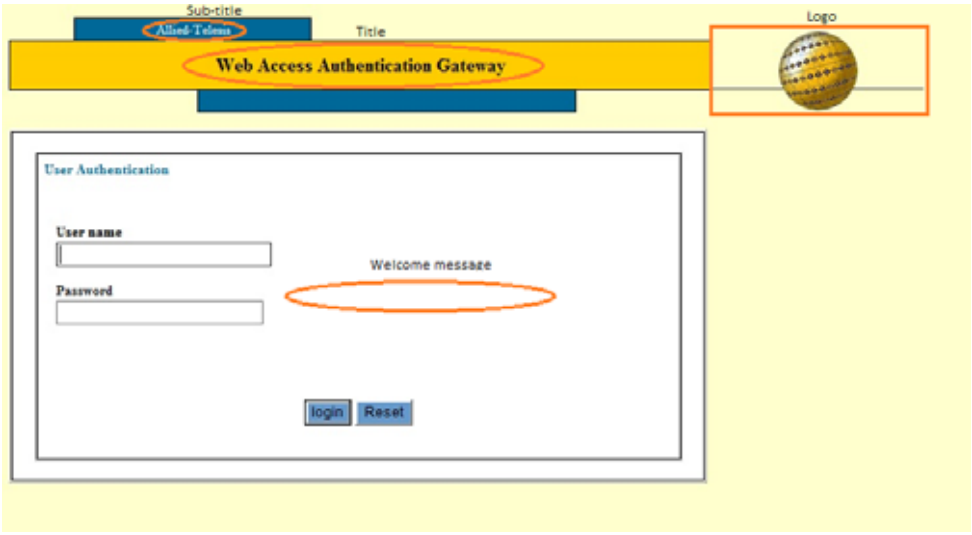
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>HTML 5 complete</title>
  </head>
  <body>
    <form action="http://<Authenticator IP address>/index.cgi"
autocomplete="off" target="_self" name="AUTH" method="POST">
      <div>User name</div>
      <div><input size="30" type="text" maxlength="64"
name="USERNAME"></div>
      <div>Password</div>
      <div><input size="30" type="password" maxlength="64"
name="PASSWORD"></div>
      <div>
        <input type="submit" name="ACTION" value="login">
        <input type="reset" name="RESET" value="Reset">
      </div>
    </form>
  <body>
</html>
```

Note that Authenticator and the external web server don't communicate with each other. The role of the external web server is just to serve a custom login page.

Also note that the external login server feature and the auth-web dhcp server feature are mutually exclusive. You cannot configure auth-web-server login-url and auth-web-server dhcp ipaddress at the same time.

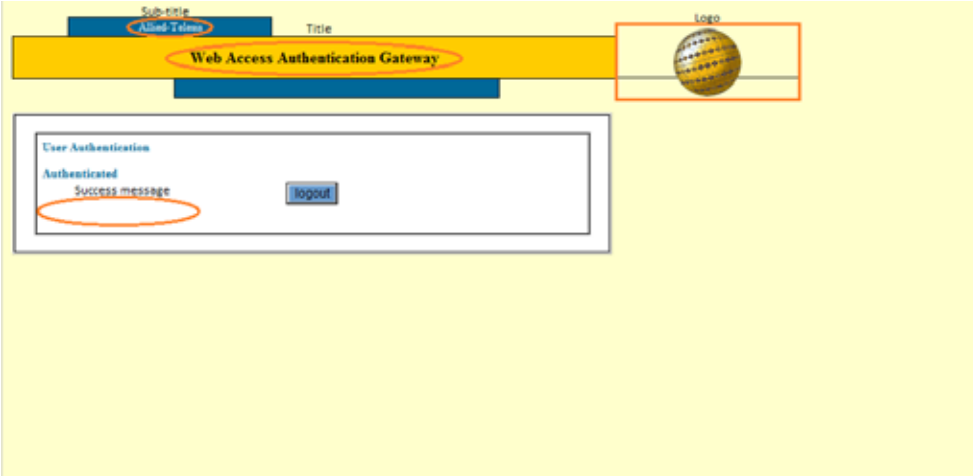
Customizing Web-Authentication Login Page

You can customize the Web-Authentication page by changing the web page logo image, success message, welcome message, web page title and sub-title. For more information



about the commands used to customize the page, see **Chapter 65, Authentication**

Commands. The elements that can be customized are shown below:



- To set the custom sub-title and title on the Web-Authentication page, use the commands below:

```
auth-web-server page sub-title {hidden|text <sub-title>}
auth-web-server page title {hidden|text <title>}
```

- To set the success and welcome message appearing on the Web-Authentication page, use the commands below:

```
auth-web-server page welcome-message text <welcome-message>
auth-web-server page success-message text <success-message>
```

To set the type of logo appearing on the Web-Authentication page, use the commands below:

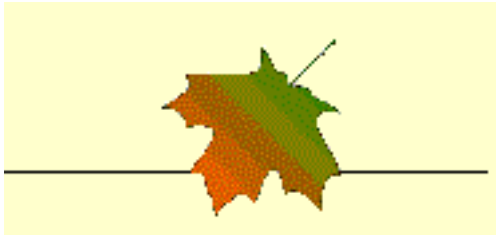
```
auth-web-server page logo {auto|default|hidden}
```

To display the custom logo if installed, use the **auto** parameter.
To display the default Allied Telesis globe logo, use the **default** parameter.
To hide the logo, use the **hidden** parameter.

You need to upload the custom logo to the flash of the device from an external storage media such as SD card, USB device or TFTP server. For example, to upload a custom logo from a TFTP server to the flash of the device, use the commands below:

```
copy tftp://<tftp server address>/my_logo.gif  flash:/logo.gif
```

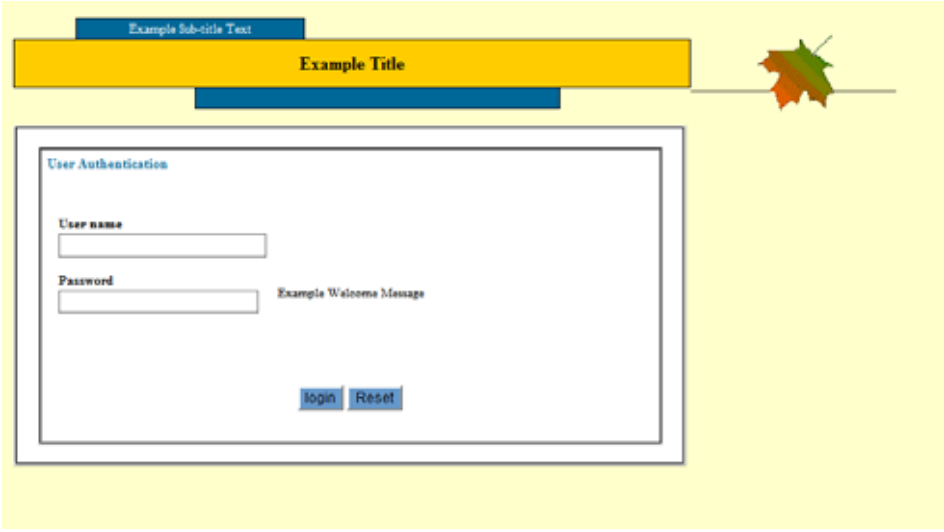
The logo file name must be logo.gif and the file format must be gif. The recommended dimension for the file is 185 X 90 pixels. The file must be stored in flash:/logo.gif on the device. An example logo is shown below:

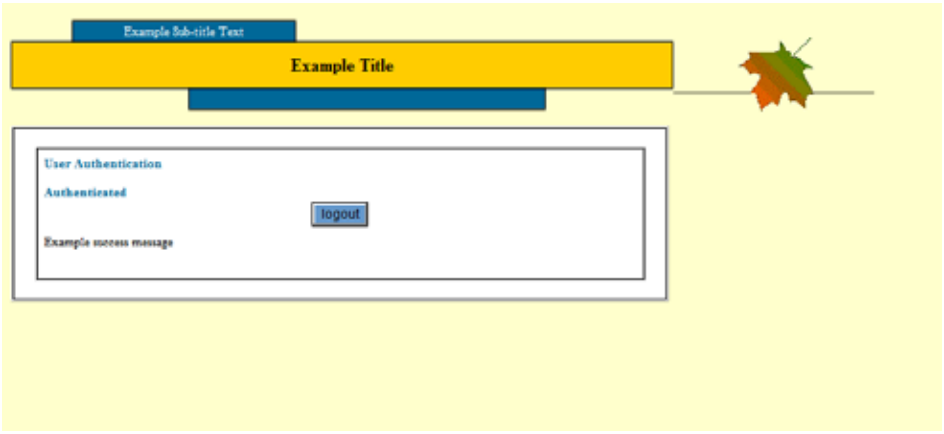


Use the following commands to customize the login page.

```
auth-web-server page sub-title text Example Sub-title Text
auth-web-server page title Example Title

auth-web-server page welcome-message text Example Welcome Message
auth-web-server page success-message text Example success Message
```

The custom Web-Authentication login page is displayed below:





Support for Protocols Underlying Web-Authentication

Web-Authentication does not use a dedicated protocol like 802.1X, with a standards-defined set of messages for authentication conversation. Instead, the switch overlays the Web-Authentication process on top of the web browser communication process. The browser communication process was not designed for authentication and is itself reliant on IP addressing, ARP, and DNS.

The authentication needs to occur in a seamless manner for all users, irrespective of their IP and DNS setting, and before they have full access to the network.

To make this possible, the switch needs to provide facilities that enable the user's PC to access the authentication web page.

Following features of Web-Authentication work together to achieve this.

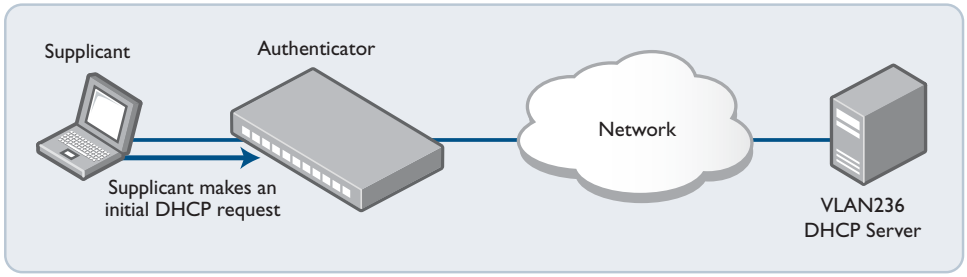
- DHCP server for Web-Authentication
- Interception of clients' ARPs
- Proxy DNS response

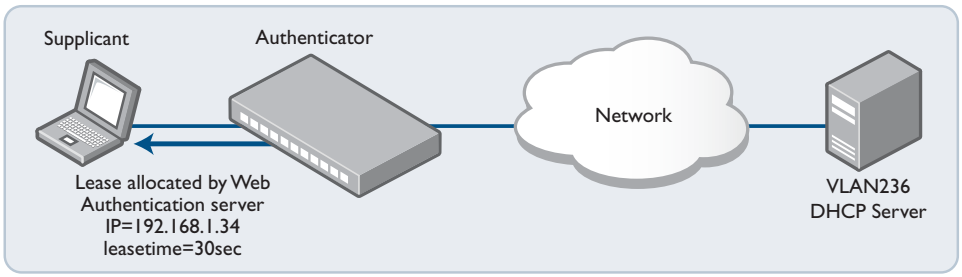
DHCP server for Web-Authentication

To initiate a web browsing session, the supplicant needs an IP address. If the supplicant has been configured to obtain its IP address by DHCP, then the authenticating switch needs to ensure that the supplicant will be served an IP address.

The simplest way to achieve this, is to have the Web-Authentication process itself act as a DHCP server. This avoids forwarding the supplicant's DHCP request to any other DHCP server. Therefore, there is a DHCP server built in to Web-Authentication.

This DHCP server is dedicated to serving IP addresses to be used by Web-Authentication clients.





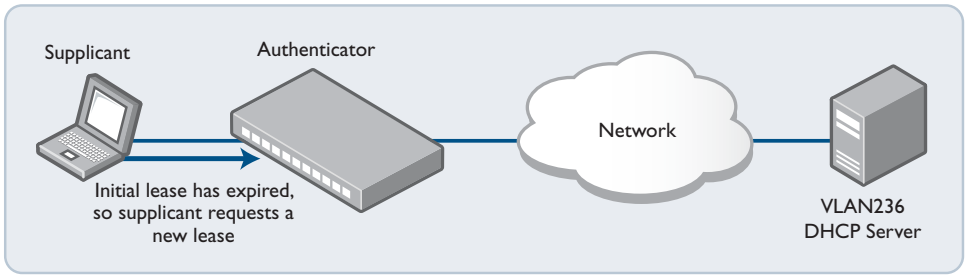
This DHCP service is configured by the command:

```
awplus(config)# auth-web-server dhcp ipaddress <ip-address/
                prefix-length>
```

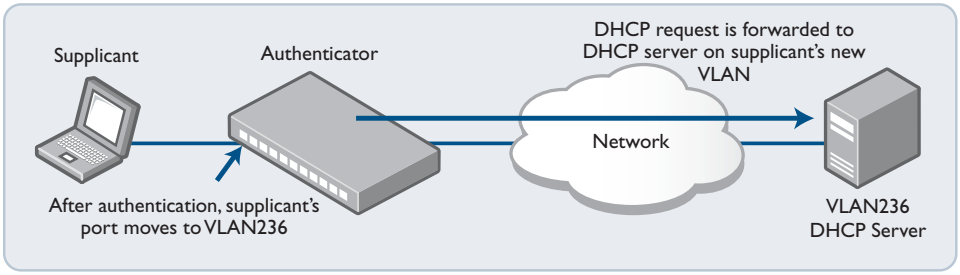
See the **auth-web-server dhcp ipaddress** command on page 65.41 and the **auth-web-server dhcp lease** command on page 65.42 for details about configuring the Web-Authentication DHCP Server.

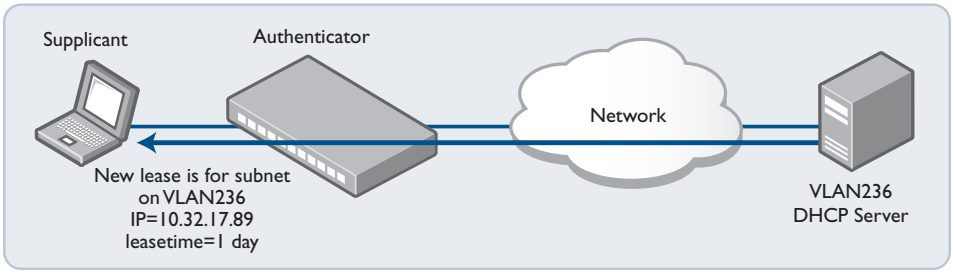
How can we force the supplicant to request a new DHCP lease after the completion of the authentication process? There is no mechanism by which the supplicant's web browser signals down to the DHCP client process to say "I've just completed an authentication session, so you need to request a new DHCP lease".

The solution is to ensure that the lease allocated by the dedicated Web-Authentication DHCP service is of a very short duration. This way the lease will expire within a short time from the completion of the authentication process, resulting in the supplicant requesting a new lease.



This new request will now be serviced by the DHCP server on the supplicant's new VLAN.

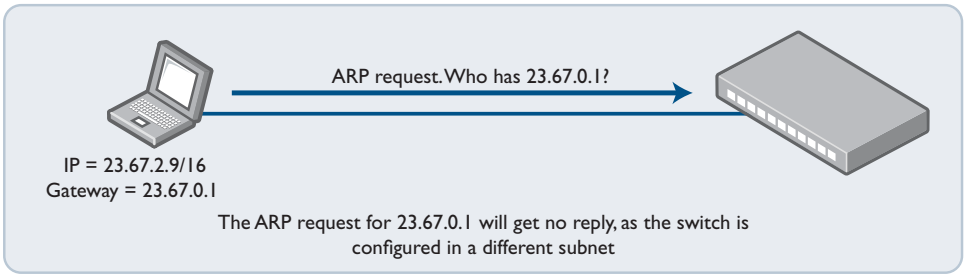




Interception of clients' ARPs

If the supplicant has been configured with a static IP address, then it is more than likely that the supplicant's IP configuration bears no relation to the Web-Authentication server address. A computer's IP communications will always be preceded by sending out ARP requests for host addresses in its local subnet, or for its gateway address.

If the IP address and gateway address have been statically configured on the computer, and the subnet used in this static configuration is different to that on the authenticating switch, then the ARP requests will receive no reply, and the PC will not begin IP communication.



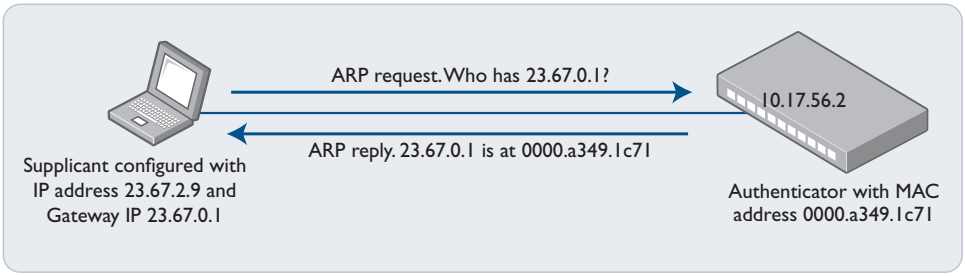
To deal with any arbitrary IP configuration on the supplicants, Web-Authentication needs a method for replying to arbitrary ARP requests. This is the ARP interception feature.

ARP interception can operate in three modes.

1. **Intercept** – will respond to ARP requests for any IP address that is in the same subnet as the switch's own IP address. Will provide its own MAC address in the ARP reply, irrespective of what IP address (within its own subnet) was being requested.

2. **None** – will only respond to ARP requests for its own IP address.

3. **Promiscuous** – will respond to **any** ARP request. Will provide its own MAC address in the ARP reply, irrespective of what IP address was being requested. When this mode is configured, the Web-Authentication server can interoperate with **any** static IP configuration on a supplicant.



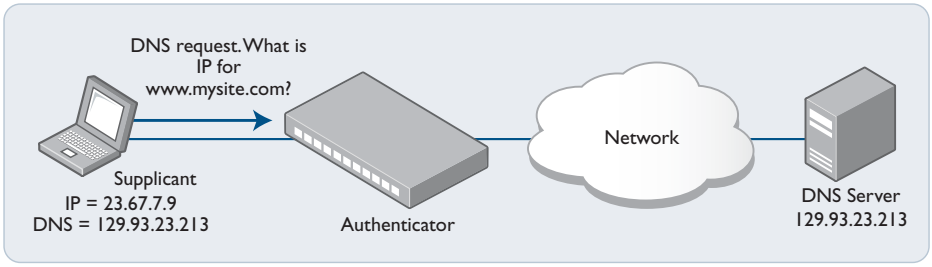
In promiscuous mode, the switch will send its own MAC address in response to an ARP request for ANY address, no matter whether the requested address bears any relation to the switch's own IP address on the interface where the ARP is received.

See the **auth-web-server mode (deleted)** command on page 65.49 for command information about setting the Web-Authentication mode.

Proxy DNS response

Typically, an HTTP session from a web browser is preceded by a DNS request for the IP address of the web site the user wishes to browse to. If the DNS request receives no reply, the web browser will never progress on to connecting an HTTP session.

The Web-Authentication server needs a mechanism to reply to DNS requests, so that the Web-Authentication session can begin.



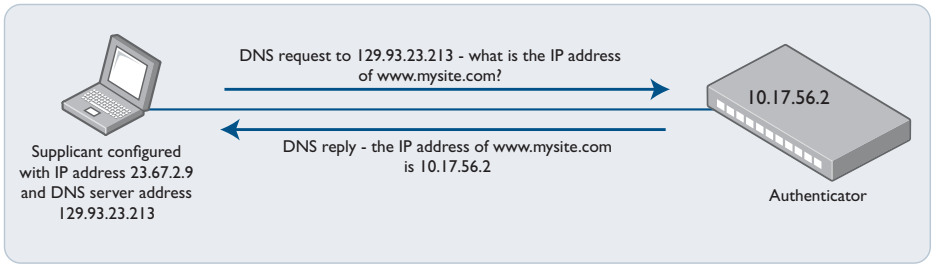
A web browser must request a DNS Server for the IP address corresponding to a URL. But the switch will not forward the request if the supplicant is not yet authenticated

The three modes listed also control the operation of the proxy DNS replies.

1. **Intercept** – responds to DNS requests whose source IP address is within the same subnet as the IP address on the switch. The IP address provided as the resolution of the DNS lookup is the switch's own IP address, so that the subsequent HTTP traffic will be directed to the switch.

2. **None** – the default. Does not respond to DNS requests.

3. **Promiscuous** – responds to DNS requests from any source IP address. The IP address provided as the resolution of the DNS lookup is the switch's own IP address, so that the subsequent HTTP traffic will be directed to the switch.



In promiscuous mode, the switch will reply to ANY DNS request from an authenticated supplicant, regardless of whether the destination IP address of the DNS server bears any relation to the switch's own IP address. The DNS reply from the switch will always specify its own IP address as the URL that was being requested.

See the **auth-web-server mode (deleted)** command on page 65.49 for command information about setting the Web-Authentication mode.

You can use the parameter <*ip-address*> of the **auth-web forward** command to specify a server for the switch to send packets from the supplicant to, for example DNS packets. For more information and an example, see the "Forwarding DNS packets using Auth-web forward command" section in the **Alliedware Plus Technical Tips and Tricks.**

Web-Authentication Timeout Connect

The command **auth timeout connect-timeout** allows you to increase the connection period for a supplicant's interface port.

The default timeout period is 30 seconds, but this command allows the timeout period to be set from 1 second to 65535 seconds. When **auth-web-server session-keep** or **auth two-step enable** is enabled, it is recommended to configure a longer connect-timeout period.

Web Authorization Proxy

Without this feature, AlliedWare Plus Web-Authentication intercepts a supplicant's initial TCP port 80 connection to a web page and sends it to the Web-Authentication Login page. However, if the supplicant is configured to use a web proxy, then it will usually be using TCP port 8080 (or another user configured port number). In this case Web-Authentication cannot intercept the connection.

To overcome this limitation, use the command **auth-web-server intercept-port**.

When a supplicant is configured to use WPAD (Web Proxy Auto-Discovery) the supplicant's web browser will use TCP port 80 as usual, and so it can be intercepted by Web-Authentication as normal, and the Web-Authentication Login page is sent. However, after authentication, it does not know where to get the WPAD file (usually named proxy.pac) that tells it what its web proxy is and so cannot access external web pages.

You can use the **auth-web-server dhcp-wpad-option** command to tell the supplicant where to find the proxy.pac file. This proxy.pac file contains the URL and/or IP address of the web proxy server that it should use.

For more information and examples, see the "Web Auth Proxy" section in the **Alliedware Plus Technical Tips and Tricks.**

MAC-Authentication

Why is MAC-Authentication Required?

The authentication mechanisms provided by 802.1X and Web-Authentication are powerful and effective. But, they are not universally applicable. Web-Authentication is only applicable to devices that have a human user who opens the web browser and types in a username and password when requested. 802.1X-Authentication is only possible from devices whose software implements an 802.1X supplicant.

There are plenty of network-connected devices, like printers, scanners, fire-alarm monitors and so on, that have neither a human user nor implement an 802.1X supplicant. In a network that ensures all access is authenticated, there needs to be a mechanism for authenticating these devices.

Fortunately, all Ethernet transceivers have a unique identifier—their MAC address. Hence, even without user input of a username and password, any Ethernet device will automatically identify itself simply by virtue of the source MAC address in the packets it sends. The method that has been developed for authenticating these devices uses the MAC address as the identifier, and so is called MAC-based authentication.

How Does MAC-Authentication Work?

In essence, MAC-authentication works little differently from 802.1X or Web-based authentication.

Here are the main steps:

1. The supplicant is connected to the switch.
2. The switch (acting as the authenticator) receives an ID from the supplicant.
3. The switch passes the supplicant's ID to a RADIUS server in an Access-Request packet
4. The RADIUS server returns an Access-Accept or an Access-Deny. The Access-Accept can be accompanied with other attributes, for dynamic VLAN assignment.

The unique aspects of MAC-Authentication are in steps 2 and 3.

MAC-Authentication does not involve a process whereby the switch sends an ID request to the supplicant. The switch receives the ID from the supplicant by simply looking at the source MAC in the packets being sent from the supplicant.

The MAC address of the supplicant is a single identifier. But a RADIUS access-request requires both a username and a password. The workaround employed by MAC-Authentication is simply to use the MAC address as both username and password.

The switch extracts the source MAC address from the supplicant's packets and puts it into a string of the form xx-xx-xx-xx-xx-xx, using lower-case letters for any hex digits in the range a-f. This string is then used as both the username and the password in the RADIUS access-request packet. The supplicant MAC address is also sent in the attribute 31 “calling-station-id” as usual.

Configuring MAC-Authentication

Under AlliedWare Plus, there are two steps to setting up MAC-Authentication.

1. Define the authentication method list that is used for MAC-Authentication.

There is only one method list that can be created for MAC-Authentication—the default method list. Moreover, the only authentication server type that can be used is RADIUS.

The command for defining the method list is:

```
awplus(config)# aaa authentication auth-mac default group
                radius
```

2. Enable MAC-Authentication on the ports that are to perform this authentication:

```
awplus(config)# interface port1.0.2
awplus(config)# auth-mac enable
awplus(config)# spanning-tree edgeport
```

On the RADIUS server, it is necessary to create user entries where both the username and password are the MAC address of the supplicant, in the form xx-xx-xx-xx-xx-xx.

For example, on the AlliedWare Plus local RADIUS server, the configuration is:

```
          awplus(config)# radius-server local
  awplus(config-radsrv)# user xx-xx-xx-xx-xx-xx
                         password xx-xx-xx-xx-xx-xx
```

The supplicant requires no configuration, as the whole purpose of MAC-Authentication is to authenticate devices that cannot be configured for authentication.

It is also possible to configure the authentication protocol that the switch uses in its interaction with the RADIUS server. There are two choices of protocol: EAP-MD5 and PAP. The default method is PAP, and can be changed by using the command:

```
awplus(config-if)# auth-mac method [eap-md5|pap]
```

Tri-Authentication

The switch supports three types of authentication for devices that connect to switch ports.

- 802.1X-Authentication of devices connecting to switch ports
- MAC-Authentication of devices connecting to switch ports
- Web-Authentication of devices connecting to switch ports

All three types can be configured to run simultaneously on a switch port. The simultaneous configuration and authentication of all three types on a port is called tri-authentication.

Tri-Authentication Configuration

Follow the below three steps to configure tri-authentication across a range of switch ports:

Step 1: Define the RADIUS Server:

Define the RADIUS Server where the switch will send authentication requests by using the below commands:

```
awplus# configure terminal
awplus(config)# radius-server host <ip-address> key
<key-string>
```

These commands adds the RADIUS Server address and set parameters to the RADIUS server. The key parameter specifies the secret key for the server.

Note The RADIUS Server, where the switch sends authentication requests, can be the switch's own Local RADIUS Server. For information on how to configure Local RADIUS Server see **Chapter 72, Local RADIUS Server Introduction and Configuration**.

Step 2: Define the default authentication server lists:

Define the default authentication server lists for 802.1X-Authentication, Web-Authentication, and MAC-Authentication:

```
awplus# configure terminal
awplus(config)# aaa authentication dot1x default group radius
awplus(config)# aaa authentication auth-web default group
radius
awplus(config)# aaa authentication auth-mac default group
radius
```

Step 3: Enable 802.1X-Authentication, Web-Authentication, and MAC-Authentication:

Follow the instructions below to enable 802.1X-Authentication, Web-Authentication, and MAC-Authentication on switch ports to attach supplicant devices. This authenticates the supplicant if any of the three methods that the supplicant tries work, depending on the auth-fail VLAN settings. After enabling the authentication, refer to earlier chapters to configure VLAN, IP address and other authentication configurations for the authentication type you want.

```
awplus# configure terminal
awplus(config)# interface <interface-range>
awplus(config-if)# switchport mode access
awplus(config-if)# switchport access vlan 1
awplus(config-if)# auth-web enable
awplus(config-if)# auth-mac enable
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth dynamic-vlan-creation
```

Two-Step Authentication

The single step authentication methods (either user or device authentication) have a potential security risk:

- an unauthorized user can access the network with an authorized device
- an authorized user can access the network with an unauthorized device

Two-step authentication solves this problem by authenticating both the user and the device. The supplicant will only become authenticated if both these steps are successful. If the first authentication step fails, then the second step is not started.

The following authentication sequences are supported for two-step authentication:

MAC-Authentication followed by 802.1X-Authentication

MAC-Authentication followed by Web-Authentication

802.1X-Authentication followed by Web-Authentication.

To configure two-step authentication:

1. Configure the first authentication method.
2. Configure the second authentication method.
3. Specify the command **auth two-step enable**.
4. Make sure that both authentication steps require different authentication credentials. See **"Ensuring Authentication Methods Require Different Usernames and Passwords" on page 64.26**.

For more information and examples, see the "Two-step authentication" section in the **Alliedware Plus Technical Tips and Tricks.**

Ensuring Authentication Methods Require Different Usernames and Passwords

If you configure a user or device to use multiple authentication methods, you need to set up your system to avoid a potential vulnerability.

The vulnerability occurs because there is no way for a RADIUS server to determine what authentication method you are using. Authentication simply queries a RADIUS server to see whether a username/password pair is valid.

This means that if you use the same RADIUS server for multiple authentication methods, a user can enter the *same* username/password pair for each of these authentication methods. If that username/password pair is valid for one of the methods, it will work for all of them.

This vulnerability is particularly significant for MAC-Authentication, because the default username and password is the MAC address of the supplicant device, which is easy to discover.

For example, if you set up two-step authentication of MAC-Authentication and 802.1X-Authentication, and both use the same RADIUS server, then an attacker does not need to know the 801.1x username and password. Instead, they can pass the 802.1X-Authentication step by entering the device's MAC address into the 802.1X username and password fields.

To avoid this vulnerability:

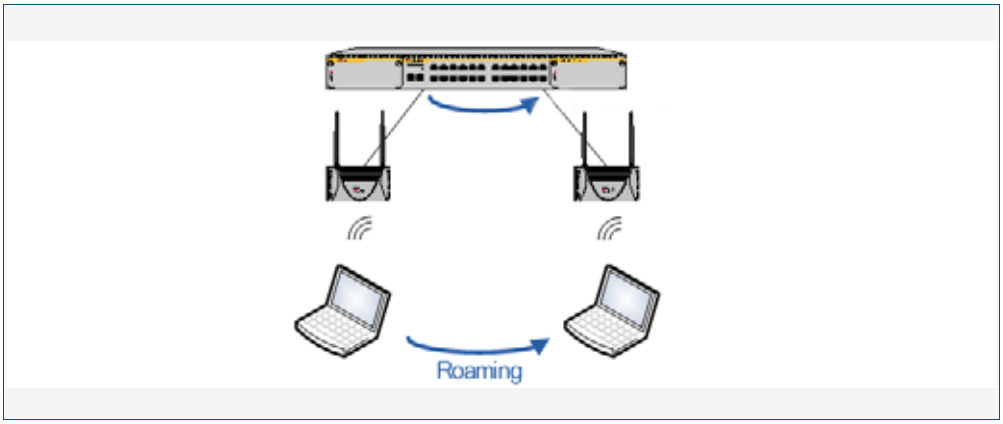
- Use different RADIUS servers for each authentication method, and/or
- Change the default password for MAC-Authentication, by using the **auth-mac password** command on page 65.33.

Roaming Authentication

When network security is required, the usability of network security must be considered. The Roaming Authentication feature improves the usability of network security by enabling users to move within the network without requiring them to re-authenticate each time they move.

If a supplicant (client device) moves from one wireless access point to another wireless access point, and the wireless access points are connected to different ports, then the switch (authenticator) recognizes that the supplicant has been authenticated and accepts the supplicant without requiring re-authentication.

Figure 64-1: Diagram showing Roaming Authentication running on a standalone switch

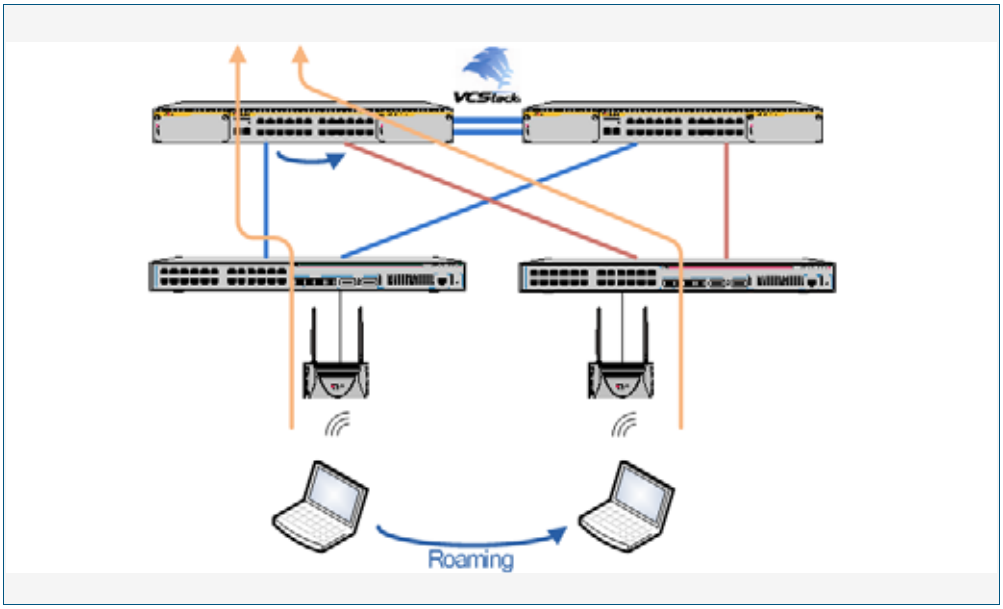


Web-Authentication and MAC-Authentication are the authentication methods in a Wireless LAN environment, and 802.1X is the authentication method used for supplicants attached to edge switches.

Roaming Authentication is normally enabled using the **auth roaming enable** command on page 65.19 command. However, Roaming Authentication has been extended (with the **auth roaming disconnected** command on page 65.17) to work where an interface is link down. This allows you to enable supplicants to move from authenticated interfaces that are link down, without requiring re-authentication.

Roaming Authentication is available for use with the VCStack feature, and is available on static and dynamic (LACP) channel group interfaces.

Figure 64-2: Diagram showing Roaming Authentication running with VCStack



Roaming Authentication Overview

If the Roaming Authentication feature is **disabled** and a supplicant moves from one switchport to another, the supplicant's authenticated status, authentication, and assigned VLAN are deleted. The supplicant is re-authenticated at the new switchport so that the network can be accessed. While the supplicant is being re-authenticated, all traffic from the supplicant is dropped.

When the Roaming Authentication feature is **enabled**, a new switch port inherits the status of a supplicant from the switch port that the supplicant moved from. If the Roaming Authentication feature is enabled on a switch, then once a supplicant (client device) is authenticated on the switch it does not have to be re-authenticated if it moves between the ports of that switch. Supplicant traffic is not dropped because there is no delay for re-authentication, during which the supplicant would be unable to access the network.

For example, when the Roaming Authentication feature is used in a wireless LAN environment with wireless access points, the wireless clients can roam between wireless access points connected to different switch ports without re-authentication.

The Roaming Authentication feature also supports VCStack operation and works on defined static channel group (static aggregators) and dynamic channel group (LACP) interfaces. When VCStack and Roaming Authentication features are used together, the status of a supplicant is inherited from one aggregated interface to another aggregated interface over the stack.

See the **auth roaming disconnected** command on page 65.17 and the **auth roaming enable** command on page 65.19 for further information about configuring Roaming Authentication.

Roaming Authentication Feature Interactions

When the Roaming Authentication feature is disabled, a supplicant must be re-authenticated on the destination interface when it roams. When the Roaming Authentication is enabled, the following supplicant authentication status and information is inherited from the source interface:

- Authentication status
- Authentication method
- Supplicant MAC address
- Supplicant IP address
  (if an authenticated interface is configured for Web-Authentication)
- Supplicant name
- Authorized dynamic VLAN ID
- Authorized RADIUS server
- Reauthentication timer
  (if configured using the **auth timeout reauth-period** command on page 65.25)

Roaming Authentication is only supported between interfaces with the same authentication configuration. If source and destination interfaces have different authentication configuration then the supplicant will be re-authenticated at the destination interface.

When the host mode is set with the **auth host-mode** command on page 65.11, a supplicant is not authenticated on a destination interface, and the authentication status is deleted on the source interface.

When a supplicant moves from an interface with authentication configured to an interface without authentication configured, the supplicant's authentication status is deleted.

A supplicant is re-authenticated when it moves to a destination interface that is configured on a different VLAN than the VLAN that is configured for the source interface.

See the following Roaming Authentication feature interactions:

- Multiple Dynamic VLANs are supported when configured with the **auth dynamic-vlan-creation** command on page 65.6 using the **multi** parameter. Multiple Dynamic VLANs are disabled by default.
- Supplicants are re-authenticated on the destination interface if the VLAN ID changes when Single Dynamic VLANs are configured with the **auth dynamic-vlan-creation** command on page 65.6 the using the **single** parameter. Single Dynamic VLANs are disabled by default.
- The Roaming Authentication feature is supported on Guest VLANs configured by the **auth guest-vlan** command on page 65.9.

When the Roaming Authentication feature is configured for use on a stack with the VCStack feature, note that supplicants are initialized and re-authenticated if a VCStack failover occurs.

Unauthenticated Supplicant Traffic

When any authentication is configured on a switch port, the question arises as to what the switch does with packets that arrive into the switch port from unauthenticated supplicants.

Unauthenticated supplicants fall into three categories listed below:

- Newly attached supplicants, which are still in the process of their first authentication attempt
- Supplicants that have made an authentication attempt, but have failed authentication
- Supplicants that have been attached, but have not made an authentication attempt. For example, on a port that has only 802.1X-Authentication enabled, any supplicant that has no 802.1X client software will not be able to attempt 802.1X-Authentication.

In switches that are running the AlliedWare Plus™ Operating System, packets from all these three categories of unathenticated supplicants are treated equally; no distinction is made between these three categories. The treatment of the traffic from unauthenticated supplicants does, however, depend on two factors:

- Whether a Guest VLAN has been configured on the switch port to which the supplicant is attached
- Whether Web-Authentication has been configured on the switch port to which the supplicant is attached

The rules governing the treatment of packets from unauthenticated supplicants are laid out in the table below:

Table 64-1: Treatment of packets from unauthenticated supplicants

Switch port configuration	No Guest VLAN configured	No Guest VLAN configured, auth-fail VLAN configured	Guest VLAN configured
Web-Authentication configured	Packets from unauthenticated supplicants are associated with the Native VLAN of the port. Packets from unauthenticated supplicants are processed according these rules: ■ Packets destined to the WebAuth server IP address/ TCP port are forwarded to the server (which may well be the switch itself). ■ DHCP packets are sent to the CPU, to be processed by a local DHCP server, or relayed to another DHCP server, depending on the configuration of the switch. ■ DNS packets are forwarded to the CPU, and then sent on to a DNS server, if the switch is configured with a DNS server address. ■ ARP packets are forwarded to the CPU, and an ARP entry for the supplicant is learned. ■ If web-auth forwarding is enabled for particular types of packets, then those packets will be forwarded within the Native VLAN ■ All other packets are dropped.	Packets from unauthenticated supplicants are associated with the Native VLAN of the port. If newly connected supplicants attempt 802.1X port authentication or Web-Authentication and fail, then they are moved to the auth-fail VLAN.	Packets from unauthenticated supplicants are associated with the Guest VLAN of the port. Packets from unauthenticated supplicants are processed according to these rules: ■ Packets destined to the WebAuth server IP address/ TCP port are forwarded to the server (which may well be the switch itself). ■ DHCP packets are sent to the CPU, to be processed by a local DHCP server, or relayed to another DHCP server, depending on the configuration of the switch. ■ DNS packets are forwarded to the CPU, and then sent on to a DNS server, if the switch is configured with a DNS server address. ■ ARP packets are forwarded to the CPU, and an ARP entry for the supplicant is learned. ■ Drop all other packets destined to the IP address of the Guest VLAN. ■ Layer 2 forward packets destined to other addresses within the Guest VLAN. ■ All other packets are dropped.
No Web-Authentication configured	All non-eap packets from unauthenticated supplicants are dropped.	All non-eap packets from unauthenticated supplicants are dropped.	Packets from unauthenticated supplicants are associated with the Guest VLAN of the port. The packets are processed according to these rules: ■ Drop packets destined to the IP address of the Guest VLAN. ■ Layer 2 forward packets destined to other addresses within the Guest VLAN. ■ Drop all other packets.

Deciding When a Supplicant Fails Authentication

Although the treatment of packets from unauthenticated supplicants does not differentiate between the three categories of supplicant, it is still useful to know for sure when the switch decides that a supplicant has failed authentication.

The rules for deciding that a supplicant has failed authentication are listed below for each type of authentication available:

Deciding when a supplicant fails 802.1X-Authentication

If the supplicant responds to EAP authentication requests, and the supplicant's authentication information is sent to the RADIUS server, and the RADIUS server replies with an Authentication-Reject, then the supplicant is immediately deemed to have failed authentication.

If the supplicant does not respond to EAP authentication requests, then the switch will resend the authentication requests up to a maximum number of attempts set by the command **dot1x max-reauth-req** (the default is 2). The interval between the attempts is set by the command **dot1x timeout tx-period** (the default is 30 seconds). If the supplicant still has not responded after this, it is deemed to have not attempted authentication.

See **Chapter 69, 802.1X Commands** for 802.1X-Authentication command information.

Deciding when a supplicant fails Web-Authentication

As soon as the supplicant attempts any web-browsing, the switch will intercept the web session, and present the supplicant with an authentication request page. If the user enters a username and password, and clicks the login button, then the switch will send the username and password to the RADIUS server. If the RADIUS server replies with an Authentication-Reject, then the supplicant is immediately deemed to have failed authentication.

Until the supplicant has attempted any web-browsing, or has received the authentication request page, but not yet clicked the login button, the supplicant is deemed to be not yet authenticated (as against not able to authenticate).

See **Chapter 65, Authentication Commands** for Web-Authentication command information.

Deciding when a supplicant fails MAC-Authentication

As soon as the supplicant sends any packet, the source MAC address from the packet will be sent to the RADIUS server for authentication. If the RADIUS server replies with an Authentication-Reject, then the supplicant is immediately deemed to have failed authentication.

With MAC-Authentication there really is no concept of not-yet-attempted authentication, because authentication is attempted as soon as a supplicant sends a packet.

See **Chapter 65, Authentication Commands** for MAC-Authentication command information.

Failed Authentication VLAN

The auth-fail VLAN feature allows the network administrator to separate the supplicants who attempted authentication, but failed, from the supplicants who did not attempt authentication.

This feature enables the network administrator to enact a security policy in which the supplicants who fail authentication are given extremely limited access, or are given access to remedial applications.

If the Guest VLAN and auth-fail VLAN are both configured on a switch, then a newly connected supplicant initially belongs to the Guest VLAN. If newly connected supplicants attempt 802.1X port authentication or Web-Authentication and fail, then they are moved from the Guest VLAN to the auth-fail VLAN.

The criteria for how many failed authentication attempts are allowed before the supplicant is moved to the auth-fail VLAN differs, depending on the authentication method used.

If Web-Authentication is used, then the supplicant is moved to the auth-fail VLAN after the first failed attempt. If 802.1X port authentication is used, then the supplicant is moved to the auth-fail VLAN after the number of failed attempts is equal to the value configured by the dot1x max-auth-fail command (by default, three failed 802.1X-Authentication attempts are allowed).

The MAC-Authentication feature does not support the max-auth-fail option. If auth-fail VLAN feature is used in conjunction with MAC-Authentication only one attempt is allowed for a MAC-Authentication supplicant. If the attempt fails, then the supplicant will be treated as "Authenticated" and the interface will be added to the configured auth-fail VLAN.

Limitations on Allowed Feature Combinations

Note that the Web-Authentication feature cannot be used with the Guest VLAN or auth-fail VLAN features. For further limitation information see the below tables:

Table 64-2: Interoperation of authentication types with guest VLAN and auth-fail VLAN

Authentication Type	Guest VLAN (without routing mode)	Guest VLAN (with routing mode)	Failed Authentication VLAN
802.1X-Authentication	Layer 2 forward packets destined to other addresses within the Guest VLAN.	Unauthorized supplicant can access Guest VLAN. Use ACL for security on the interface.	Failed authentication supplicant can access auth-fail VLAN. See limitations table below for ACL usage limitation.
MAC-Authentication	Layer 2 forward packets destined to other addresses within the Guest VLAN.	Unauthorized supplicant can access Guest VLAN. Use ACL for security on the interface.	Failed authentication supplicant can access auth-fail VLAN. See limitations table below for ACL usage limitation.
Web-Authentication (without intercept mode)	Layer 2 forward packets destined to other addresses within the Guest VLAN.	Unauthorized supplicant can access Guest VLAN. Use ACL for security on the interface.	Failed authentication supplicant can access auth-fail VLAN. See limitations table below for ACL usage limitation.
Web-Authentication (with intercept mode)	(Not Available)	(Not Available)	(Not Available)

Table 64-3: Interactions between guest VLAN and auth-fail VLAN

Authentication Feature	Guest VLAN (without routing mode)	Guest VLAN (with routing mode)	Failed Authentication VLAN
Guest VLAN (without routing mode)	(Not Available)	(Not Available)	Cannot configure ACLs on the Guest VLAN when it is not in routing mode. The Guest VLAN without routing mode has reserved ACLs already attached to it.
Guest VLAN (with routing mode)	(Not Available)	(Not Available)	Configuration of ACLs for additional interface security is recommended.
Failed Authentication VLAN	Cannot configure ACLs on the Guest VLAN when it is not in routing mode. The Guest VLAN without routing mode has reserved ACLs already attached to it.	Configuration of ACLs for additional interface security is recommended.	Failed authentication supplicant can access auth-fail VLAN. See limitations table below for ACL usage limitation.

Chapter 65: Authentication Commands

Command List 65.3
auth auth-fail vlan 65.3
auth critical 65.5
auth dynamic-vlan-creation 65.6
auth guest-vlan 65.9
auth host-mode 65.11
auth log 65.13
auth max-supplicant 65.15
auth reauthentication 65.16
auth roaming disconnected 65.17
auth roaming enable 65.19
auth supplicant-mac 65.21
auth timeout connect-timeout 65.23
auth timeout quiet-period 65.24
auth timeout reauth-period 65.25
auth timeout server-timeout 65.26
auth timeout supp-timeout 65.27
auth two-step enable 65.28
auth-mac enable 65.31
auth-mac method 65.32
auth-mac password 65.33
auth-mac reauth-relearning 65.34
auth-web enable 65.35
auth-web forward 65.36
auth-web max-auth-fail 65.38
auth-web method 65.39
auth-web-server blocking-mode 65.40
auth-web-server dhcp ipaddress 65.41
auth-web-server dhcp lease 65.42
auth-web-server dhcp-wpad-option 65.43
auth-web-server gateway (deleted) 65.44
auth-web-server host-name 65.45
auth-web-server http-redirect (deleted) 65.45
auth-web-server intercept-port 65.46
auth-web-server ipaddress 65.47
auth-web-server login-url 65.48
auth-web-server mode (deleted) 65.49
auth-web-server page logo 65.49
auth-web-server page sub-title 65.50
auth-web-server page success-message 65.51
auth-web-server page title 65.52
auth-web-server page welcome-message 65.53
auth-web-server ping-poll enable 65.54
auth-web-server ping-poll failcount 65.55
auth-web-server ping-poll interval 65.56
auth-web-server ping-poll reauth-timer-refresh 65.57
auth-web-server ping-poll timeout 65.58

auth-web-server port 65.59
auth-web-server redirect-delay-time 65.60
auth-web-server redirect-url 65.61
auth-web-server session-keep 65.62
auth-web-server ssl 65.63
auth-web-server sslport (deleted) 65.63
auth-web-server ssl intercept-port 65.64
copy proxy-autoconfig-file 65.65
copy web-auth-https-file 65.65
erase proxy-autoconfig-file 65.66
erase web-auth-https-file 65.66
platform l3-vlan-hashing-algorithm 65.67
platform mac-vlan-hashing-algorithm 65.68
show auth two-step supplicant brief 65.69
show auth-mac 65.70
show auth-mac diagnostics 65.71
show auth-mac interface 65.72
show auth-mac sessionstatistics 65.75
show auth-mac statistics interface 65.76
show auth-mac supplicant 65.77
show auth-mac supplicant interface 65.79
show auth-web 65.80
show auth-web diagnostics 65.82
show auth-web interface 65.83
show auth-web sessionstatistics 65.86
show auth-web statistics interface 65.87
show auth-web supplicant 65.88
show auth-web supplicant interface 65.89
show auth-web-server 65.90
show auth-web-server page 65.91
show proxy-autoconfig-file 65.92

Command List

This chapter provides an alphabetical reference for authentication commands.

auth auth-fail vlan

Use this command to enable the **auth-fail vlan** feature on the specified vlan interface. This feature assigns supplicants (client devices), which have failed port authentication, to the specified VLAN interface.

Use the **no** variant of this command to disable the **auth-fail vlan** feature for a specified VLAN interface.

Syntax

```
auth auth-fail vlan <1-4094>
no auth auth-fail vlan
```

Parameter	Description
<1-4094>	Assigns the VLAN ID to any supplicants that have failed port authentication.

Default The **auth-fail vlan** feature is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use the **auth-fail vlan** feature when using Web-Authentication instead of the Guest VLAN feature, when you need to separate networks where one supplicant (client device) requires authentication and another supplicant does not require authentication from the same interface.

This is because the DHCP lease time using the Web-Authentication feature is shorter, and the **auth fail vlan** feature enables assignment to a different VLAN if a supplicant fails authentication.

To enable the **auth-fail vlan** feature with Web Authentication, you need to set Web Authentication Server virtual IP address by using the **auth-web-server ipaddress** command on page 65.47 or the **auth-web-server dhcp ipaddress** command on page 65.41.

When using 802.1X port authentication, use a **dot1x max-auth-fail** command to set the maximum number of login attempts. Three login attempts are allowed by default for 802.1X port authentication before supplicants trying to authenticate are moved from the Guest VLAN to the auth-fail VLAN. See the **"dot1x max-auth-fail" on page 69.9** for command information.

See the section **"Failed Authentication VLAN" on page 64.33** in **Chapter 64, Authentication Introduction and Configuration** for further overview information about the auth-fail VLAN feature, which allows the Network Administrator to separate the supplicants who attempted authentication, but failed, from the supplicants who did not attempt authentication.

See the section **"Limitations on Allowed Feature Combinations" on page 64.33** for information about restrictions regarding combinations of authentication enhancements working together.

Use appropriate ACLs (Access Control Lists) on interfaces for extra security if a supplicant allocated to the designated auth-fail vlan can access the same network as a supplicant on the Guest VLAN. For more information about ACL concepts, and configuring ACLs see **Chapter 57, Access Control Lists Introduction**. For more information about ACL commands see:

- **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands**
- **Chapter 59, IPv4 Software Access Control List (ACL) Commands**
- **Chapter 60, IPv6 Hardware Access Control List (ACL) Commands**
- **Chapter 61, IPv6 Software Access Control List (ACL) Commands**

Examples

To enable **auth-fail vlan** for `port1.0.2` and assign VLAN `100`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth auth-fail vlan 100
```

To disable the **auth-fail vlan** feature for `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth auth-fail vlan
```

Validation Commands

show running-config

Related Commands

dot1x max-auth-fail
show dot1x
show dot1x interface

auth critical

This command enables the critical port feature on the interface. When the critical port feature is enabled on an interface, and all the RADIUS servers are unavailable, then the interface becomes authorized.

The **no** variant of this command disables critical port feature on the interface.

Syntax

```
auth critical
no auth critical
```

Default The critical port of port authentication is disabled.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable the critical port feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth critical
```

To disable the critical port feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth critical
```

Validation Commands

show auth-web-server
show dot1x
show dot1x interface
show running-config

auth dynamic-vlan-creation

This command enables and disables the Dynamic VLAN assignment feature.

The Dynamic VLAN assignment feature allows a supplicant (client device) to be placed into a specific VLAN based on information returned from the RADIUS server during authentication, on a given interface.

Use the **no** variant of this command to disable the Dynamic VLAN assignment feature.

Syntax

```
auth dynamic-vlan-creation [rule {deny|permit}] [type {multi|single}]
no auth dynamic-vlan-creation
```

Parameter	Description
`rule`	VLAN assignment rule.
`deny`	Deny a differently assigned VLAN ID. This is the default rule.
`permit`	Permit a differently assigned VLAN ID.
`type`	Specifies whether multiple different VLANs can be assigned to supplicants (client devices) attached to the port, or whether only a single VLAN can be assigned to supplicants on the port.
`multi`	Multiple Dynamic VLAN.
`single`	Single Dynamic VLAN.

Default By default, the Dynamic VLAN assignment feature is disabled.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

If the Dynamic VLAN assignment feature is enabled (disabled by default), VLAN assignment is dynamic. If the Dynamic VLAN assignment feature is disabled then RADIUS attributes are ignored and configured VLANs are assigned to ports. Dynamic VLANs may be associated with authenticated MAC addresses if the **type** parameter is applied with the **rule** parameter.

The **rule** parameter deals with the case where there are multiple supplicants attached to a port, and the type parameter has been set to **single-vlan**. The parameter specifies how the switch should act if different VLAN IDs end up being assigned to different supplicants. The keyword value **deny** means that once a given VID has been assigned to the first supplicant, then if any subsequent supplicant is assigned a different VID, that supplicant is rejected. The keyword value **permit** means that once a given VID has been assigned to the first supplicant, then if any subsequent supplicant is assigned a different VID, that supplicant is accepted, but it is actually assigned the same VID as the first supplicant.

If you issue an **auth dynamic-vlan-creation** command without an optional **rule** parameter and a required **deny** or **permit** keyword value then a second supplicant with a different VLAN ID is rejected. It is not assigned to the first supplicant's VLAN. Issuing an a**uth dynamic-vlan-creation** command without an optional **rule** parameter has the same effect as issuing an **auth dynamic-vlan-creation rule deny** command rejecting supplicants with differing VIDs.

The **type** parameter specifies whether multiple different VLANs can be assigned to supplicants attached to the port, or whether only a single VLAN can be assigned to supplicants on the port. The **type** parameter can select the port base VLAN or the MAC base VLAN from the RADIUS VLAN ID. This can be used when the host-mode is set to multi-supplicant. For **single**-host ports, the VLAN ID will be assigned to the port. It is not supported with the Guest VLAN feature. Display the ID assigned using a **show vlan** command. For **multi**-host ports, the VLAN ID will be assigned to the MAC address of the authenticated supplicant. The VLAN ID assigned for the MAC Base VLAN is displayed using the **show platform table vlan** command.

To configure Dynamic Vlan with Web Authentication, you need to set Web Authentication Server virtual IP address by using the **auth-web-server ipaddress** command on page 65.47 or the **auth-web-server dhcp ipaddress** command on page 65.41. You also need to create a hardware access-list that can be applied to the switch port interface.

You need to configure an IPv4 address for the VLAN interface on which Web Authentication is running.

Examples To enable the Dynamic VLAN assignment feature on interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport access vlan 10
awplus(config-if)# auth-web enable
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# interface vlan10
awplus(config-if)# ip address 10.1.1.1/24
```

To enable the Dynamic VLAN assignment feature with Web Authentication on interface `port1.0.2` when Web Authentication is needed, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ipaddress 1.2.3.4
awplus(config)# access-list hardware acl-web send-
to-cpu ip any 1.2.3.4
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web enable
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# access-group acl-web
awplus(config-if)# interface vlan1
awplus(config-if)# ip address 10.1.1.1/24
```

To disable the Dynamic VLAN assignment feature on interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth dynamic-vlan-creation
```

Validation Commands

show dot1x
show dot1x interface
show running-config

Related Commands

auth host-mode

auth guest-vlan

This command enables and configures the Guest VLAN feature on the interface specified by associating a Guest VLAN with an interface. This command does not start authentication. The supplicant's (client device's) traffic is associated with the native VLAN of the interface if its not already associated with another VLAN. The **routing** option enables routing from the Guest VLAN to another VLAN, so the switch can lease DHCP addresses and accept access to a limited network.

The **no** variant of this command disables the guest vlan feature on the interface specified.

Syntax

```
auth guest-vlan <1-4094> [routing]
no auth guest-vlan [routing]
```

Parameter	Description
<1-4094>	VLAN ID (VID).
`routing`	Enables routing from the Guest VLAN to other VLANs.

Default The Guest VLAN authentication feature is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage The Guest VLAN feature may be used by supplicants (client devices) that have not attempted authentication, or have failed the authentication process. Note that if a port is in multi-supplicant mode with per-port dynamic VLAN configuration, after the first successful authentication, subsequent hosts cannot use the guest VLAN due to the change in VLAN ID. This may be avoided by using per-user dynamic VLAN assignment.

When using the Guest VLAN feature with the multi-host mode, a number of supplicants can communicate via a guest VLAN before authentication. A supplicant's traffic is associated with the native VLAN of the specified switch port. The supplicant must belong to a VLAN before traffic from the supplicant can be associated.

Note that you must first define the VLAN with the **vlan** command that you will assign as a guest VLAN using this command. Also note that 802.1X must first be enabled on the port.

Guest VLAN authentication cannot be enabled if DHCP snooping is enabled (**service dhcp-snooping** command on page 80.23), and vice versa.

The Guest VLAN feature in previous releases had some limitations that have been removed. Until this release the Guest VLAN feature could not lease the IP address to the supplicant using DHCP Server or DHCP Relay features unless Web-Authentication was also applied. When using NAP authentication, the supplicant should have been able to log on to a domain controller to gain certification, but the Guest VLAN would not accept access to another VLAN.

The Guest VLAN routing mode in this release overcomes these issues. With the Guest VLAN routing mode, the switch can lease DHCP addresses and accept access to a limited network.

Note that Guest VLAN can use only untagged ports and tagged ports cannot be used for Guest VLAN.

See the section **"Configuring a Guest VLAN" on page 64.2** for information about the Guest VLAN feature.

See the section **"Limitations on Allowed Feature Combinations" on page 64.33** for information about restrictions regarding combinations of authentication enhancements working together.

Examples

To define `vlan100` and assign the guest VLAN feature to `vlan100` on interface `port1.0.2`, and enable routing from the guest vlan to other VLANs, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 100
awplus(config-vlan)# exit
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth guest-vlan 100 routing
```

To disable the guest vlan feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth guest-vlan
```

Validation Commands

show dot1x
show dot1x interface
show running-config

Related Commands

dot1x port-control
vlan

auth host-mode

This command selects host mode on the interface. Multi-host is an extension to IEEE802.1X.

Use the **no** variant of this command to set host mode to the default setting (single host).

Syntax

```
auth host-mode {single-host|multi-host|multi-supplicant}
no auth host-mode
```

Parameter	Description
`single-host`	Single host mode. In this mode, only one host may be authorized with the port. If other hosts out the interface attempt to authenticate, the authenticator blocks the attempt.
`multi-host`	Multi host mode. In this mode, multiple hosts may be authorized with the port; however only one host must be successfully authenticated at the Authentication Server for all hosts to be authorized with the port. Upon one host being successfully authenticated (state Authenticated), the other hosts will be automatically authorized at the port (state ForceAuthorized). If no host is successfully authenticated, then all hosts are not authorized with the port.
`multi-supplicant`	Multi supplicant (client device) mode. In this mode, multiple hosts may be authorized with the port, but each host must be individually authenticated with the Authentication Server to be authorized with the port. Supplicants which are not authenticated are not authorized with the port, while supplicants which are successfully authenticated are authorized with the port.

Default The default host mode for port authentication is for a single host.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Ports residing in the unauthorized state for host(s) or supplicant(s), change to an authorized state when the host or supplicant has successfully authenticated with the Authentication Server.

When multi-host mode is used or auth critical feature is used, all hosts do not need to be authenticated.

Examples To set the host mode to multi-supplicant on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth host-mode multi-supplicant
```

To set the host mode to default (single host) on interface `port1.0.2`, use the following

commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth host-mode
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth log

Use this command to configure the types of authentication feature log messages that are output to the log file.

Use the **no** variant of this command to remove either specified types or all types of authentication feature log messages that are output to the log file.

Syntax

```
auth log {dot1x|auth-mac|auth-web} {success|failure|logoff|all}
no auth log {do1x|auth-mac|auth-web} {success|failure|logoff|all}
```

Parameter	Description
`dot1x`	Specify only 802.1X-Authentication log messages are output to the log file.
`auth-mac`	Specify only MAC-Authentication log messages are output to the log file.
`auth-web`	Specify only Web-Authentication log messages are output to the log file.
`success`	Specify only successful authentication log messages are output to the log file.
`failure`	Specify only authentication failure log messages are output to the log file.
`logoff`	Specify only authentication log-off messages are output to the log file. Note that link down, age out and expired ping polling messages will be included.
`all`	Specify all types of authentication log messages are output to the log file Note that this is the default behavior for the authentication logging feature.

Default All types of authentication log messages are output to the log file by default.

Mode Interface Configuration

Examples To configure the logging of MAC-Authentication failures to the log file for supplicants (client devices) connected to interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth log auth-mac failure
```

To configure the logging of all types of authentication log messages to the log file for supplicants (client devices) connected to interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth log all
```

Validation Commands **show running-config**

auth max-supplicant

This command sets the maximum number of supplicants (client devices) on the interface that can be authenticated. After this value is exceeded supplicants are not authenticated.

The **no** variant of this command resets the maximum supplicant number to the default (1024).

Syntax

```
auth max-supplicant <2-1024>
no auth max-supplicant
```

Parameter	Description
<2-1024>	Limit number.

Default The max supplicant of port authentication is 1024.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the maximum number of supplicants to `10` on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth max-supplicant 10
```

To reset the maximum number of supplicant to default on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth max-supplicant
```

Validation Commands
show dot1x
show dot1x interface
show running-config

auth reauthentication

This command enables re-authentication on the interface specified in the Interface mode, which may be a static channel group (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the **no** variant of this command to disables reauthentication on the interface.

Syntax

```
auth reauthentication
no auth reauthentication
```

Default Reauthentication of port authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable reauthentication on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth reauthentication
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth roaming disconnected

This command enables the Roaming Authentication feature on an authenticated interface that is link down. A supplicant (a client device) is not reauthenticated when moved between authenticated interfaces, providing both interfaces have the Roaming Authentication feature enabled before the supplicant is moved.

Use the **auth roaming enable** command before using this command. The **auth roaming disconnected** command on its own will have no effect on the operation of the switch. This command will only come into effect once the base Roaming Authentication feature is enabled, using the **auth roaming enable** command.

The **no** variant of this command disables the Roaming Authentication feature on an interface, and forces a supplicant to be reauthenticated when moving between interfaces.

See **"Web-Authentication" on page 64.4** for further information about this feature.

Syntax

```
auth roaming disconnected

no auth roaming disconnected
```

Default The Roaming Authentication `disconnected` feature is disabled by default on an interface. Authentication status for a roaming supplicant is deleted by default when an interface goes down.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command allows a supplicant to move to another authenticating interface without reauthentication, if the link is down for the interface that the supplicant is moved from.

Note that 802.1X port authentication, or MAC-Authentication, or Web-Authentication must first be enabled on an interface to use this feature. The port that the supplicant is moving to must have the same authentication configuration as the port the supplicant is moving from.

Configure **auth roaming enable** on an interface before configuring **auth roaming disconnected** if you require **auth roaming disconnected** configured on an interface for a roaming supplicant.

Roaming Authentication cannot be enabled if DHCP snooping is enabled (**service dhcp-snooping** command on page 80.23), and vice versa.

Examples To enable Roaming Authentication `disconnected` feature for `port1.0.2`, after enabling 802.1X-Authentication and enabling Roaming Authentication `enable`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth roaming enable
awplus(config-if)# auth roaming disconnected
```

To disable Roaming Authentication disconnected feature for `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth roaming disconnected
```

Validation Commands

show running-config

Related Commands

auth-mac enable
auth roaming enable
auth-web enable
dot1x port-control
show auth-mac interface
show auth-web interface
show dot1x interface

auth roaming enable

This command enables the Roaming Authentication feature on an authenticated interface that is link up. A supplicant (a client device) is not reauthenticated when moved between authenticated interfaces, providing both interfaces have the Roaming Authentication feature enabled before the supplicant is moved.

Use the **auth roaming enable** command before using **auth roaming disconnected** command. The **auth roaming disconnected** command on its own will have no effect on the operation of the switch. This command will only come into effect once the base Roaming Authentication feature is enabled, using the **auth roaming enable** command.

The **no** variant of this command disables the Roaming Authentication feature on an interface, and forces a supplicant to be reauthenticated when moving between interfaces.

See **"Web-Authentication" on page 64.4** for further information about this feature.

Syntax

```
auth roaming enable
no auth roaming enable
```

Default The Roaming Authentication enable feature is disabled by default on an interface.

Authentication status for a roaming supplicant is deleted by default when an interface goes down.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command allows a supplicant to move to another authenticating interface without reauthentication, providing the link is up for the interface that the supplicant is moved from.

Note that 802.1X port authentication, or MAC-Authentication, or Web-Authentication must first be enabled on an interface to use this feature. The port that the supplicant is moving to must have the same authentication configuration as the port the supplicant is moving from.

Configure **auth roaming enable** on an interface before configuring **auth roaming disconnected** if you require **auth roaming disconnected** configured on an interface for a roaming supplicant.

Roaming Authentication cannot be enabled if DHCP snooping is enabled (**service dhcp-snooping** command on page 80.23), and vice versa.

Examples To enable the Roaming Authentication enable feature for interface `port1.0.4`, after enabling 802.1X-Authentication, since an authentication method is required, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth roaming enable
```

To disable Roaming Authentication enable for `port1.0.4`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# no auth roaming enable
```

Validation Commands

show running-config

Related Commands

auth-mac enable
auth roaming disconnected
auth-web enable
dot1x port-control
show auth-mac interface
show auth-web interface
show dot1x interface

auth supplicant-mac

This command adds a supplicant (client device) MAC address on a given interface with the parameters as specified in the table below.

Use the **no** variant of this command to delete the supplicant MAC address added by the **auth supplicant-mac** command, and resets to the default for the supplicant parameter.

Syntax

```
auth supplicant <mac-addr> [max-reauth-req <1-10>] [port-control
   {auto | force-authorized | force-unauthorized | skip-second-auth}]
   [quiet-period <1-65535>] [reauth-period <1-4294967295>]
   [supp-timeout <1-65535>]
   [server-timeout <1-65535>][reauthentication]

no auth supplicant-mac <macadd> [reauthentication]
```

Parameter	Description
<mac-addr>	MAC (hardware) address of the Supplicant entry in HHHH.HHHH.HHHH MAC address hexadecimal format.
`port-control`	Port control commands.
`auto`	Allow port client to negotiate authentication.
`force-authorized`	Force port state to authorized.
`force-unauthorized`	Force port state to unauthorized.
`skip-second-auth`	Skip the second authentication.
`quiet-period`	Quiet period in the HELD state (default 60 seconds).
<1-65535>	Seconds for quiet period.
`reauth-period`	Seconds between reauthorization attempts (default 3600 seconds).
<1-4294967295>	Seconds for reauthorization attempts (reauth-period).
`supp-timeout`	Supplicant response timeout (default 30 seconds).
<1-65535>	Seconds for supplicant response timeout.
`server-timeout`	Authentication server response timeout (default 30 seconds).
<1-65535>	Seconds for authentication server response timeout.
`reauthentication`	Enable reauthentication on a port.
`max-reauth-req`	No of reauthentication attempts before becoming unauthorized (default 2).
<1-10>	Count of reauthentication attempts.

Default No supplicant MAC address for port authentication exists by default until first created with the **auth supplicant-mac** command. The defaults for parameters applied are as shown in the table.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To add the supplicant MAC address `0009.41A4.5943` to force authorized port control for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth supplicant-mac 0009.41A4.5943 port-
control force-authorized
```

To delete the supplicant MAC address `0009.41A4.5943` for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth supplicant-mac 0009.41A4.5943
```

To reset reauthentication to disable for the supplicant MAC address `0009.41A4.5943`, for interface `port1.0.2` use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth supplicant-mac 0009.41A4.5943
reauthentication
```

Validation Commands **show dot1x**
show dot1x interface
show running-config

auth timeout connect-timeout

This command sets the connect-timeout period for the interface.

Use the **no** variant of this command to reset the connect-timeout period to the default (30 seconds).

Syntax

```
auth timeout connect-timeout <1-65535>
no auth timeout connect-timeout
```

Parameter	Description
<1-65535>	Seconds.

Default The connect-timeout default is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command is used for MAC- and Web-Authentication. If the connect-timeout has lapsed and the supplicant has the state **connecting**, then the supplicant is deleted. When **auth-web-server session-keep** or **auth two-step enable** is enabled it is recommended to configure a longer connect-timeout period.

Examples To set the connect-timeout period to `3600` for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout connect-timeout 3600
```

To reset the connect-timeout period to the default (30 seconds) for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout connect-timeout
```

Validation Commands **show dot1x**
show dot1x interface

auth timeout quiet-period

This command sets the time period for which the authentication request is not accepted on a given interface, after the authentication request has failed an authentication.

Use the **no** variant of this command to reset quiet period to the default (60 seconds).

Syntax

```
auth timeout quiet-period <1-65535>
no auth timeout quiet-period
```

Parameter	Description
<1-65535>	Seconds.

Default The quiet period of port authentication is 60 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the quiet period to `10` for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout quiet-period 10
```

To reset the quiet period to the default (60 seconds) for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout quiet-period
```

auth timeout reauth-period

This command sets the timer for reauthentication on a given interface. The re-authentication for the supplicant (client device) is executed at this timeout. The timeout is only applied if the **auth reauthentication** command is applied.

Use the **no** variant of this command to reset the **reauth-period** parameter to the default (3600 seconds).

Syntax

```
auth timeout reauth-period <1-4294967295>
no auth timeout reauth-period
```

Parameter	Description
<1-4294967295>	Seconds.

Default The default reauthentication period for port authentication is 3600 seconds, when reauthentication is enabled on the port.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the reauthentication period to 1 day for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout reauth-period 86400
```

To reset the reauthentication period to the default (3600 seconds) for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout reauth-period
```

Validation Commands **show dot1x**
show dot1x interface
show running-config

Related Commands **auth reauthentication**

auth timeout server-timeout

This command sets the timeout for the waiting response from the RADIUS server on a given interface.

The **no** variant of this command resets the server-timeout to the default (30 seconds).

Syntax

```
auth timeout server-timeout <1-65535>
no auth timeout server-timeout
```

Parameter	Description
<1-65535>	Seconds.

Default The server timeout for port authentication is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the server timeout to 120 seconds for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout server-timeout 120
```

To set the server timeout to the default (30 seconds) for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout server-timeout
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth timeout supp-timeout

This command sets the timeout of the waiting response from the supplicant (client device) on a given interface.

The **no** variant of this command resets the supplicant timeout to the default (30 seconds).

Syntax

```
auth timeout supp-timeout <1-65535>
no auth timeout supp-timeout
```

Parameter	Description
<1-65535>	Seconds.

Default The supplicant timeout of port authentication is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the server timeout to `2` seconds for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth timeout supp-timeout 2
```

To reset the server timeout to the default (30 seconds) for interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth timeout supp-timeout
```

Validation Commands

show dot1x
show dot1x interface
show running-config

auth two-step enable

This command enables a two-step authentication feature on an interface. When this feature is enabled, the supplicant is authorized in a two-step process. If authentication succeeds, the supplicant becomes authenticated. This command will apply the two-step authentication method based on 802.1X-, MAC- or Web-Authentication.

The **no** variant of this command disables the two-step authentication feature.

Syntax

```
auth two-step enable
no auth two-step enable
```

Default Default.

Mode Interface Configuration for a port.

Usage The single step authentication methods (either user or device authentication) have a potential security risk:

- an unauthorized user can access the network with an authorized device, or
- an authorized user can access the network with an unauthorized device.

Two-step authentication solves this problem by authenticating both the user and the device. The supplicant will only become authenticated if both these steps are successful. If the first authentication step fails, then the second step is not started.

Examples To enable the two step authentication feature, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth two-step enable
```

To disable the two step authentication feature, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth two-step enable
```

To enable MAC-Authentication followed by 802.1X-Authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode access
awplus(config-if)# auth-mac enable
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# auth two-step enable
```

To enable MAC-Authentication followed by Web-Authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode access
awplus(config-if)# auth-mac enable
awplus(config-if)# auth-web enable
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# auth two-step enable
```

To enable 802.1X-Authentication followed by Web-Authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode access
awplus(config-if)# auth-web enable
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# auth two-step enable
```

Validation Commands

show startup-config
show auth-mac supplicant
show dot1x supplicant

Related Commands

show auth two-step supplicant brief
show auth-mac
show auth-mac interface
show auth-mac supplicant
show auth-web
show auth-web interface
show auth-web supplicant
show dot1x
show dot1x interface
show dot1x supplicant

auth-mac enable

This command enables MAC-based authentication on the interface specified in the Interface command mode.

Use the **no** variant of this command to disable MAC-based authentication on an interface.

Syntax

```
auth-mac enable
no auth-mac enable
```

Default MAC-Authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Enabling **spanning-tree edgeport** on ports after enabling MAC-based authentication avoids unnecessary re-authentication when the port state changes, which does not happen when spanning tree edgeport is enabled. Note that re-authentication is correct behavior without **spanning-tree edgeport** enabled.

Applying **switchport mode access** on ports is also good practice to set the ports to access mode with ingress filtering turned on, whenever ports for MAC-Authentication are in a VLAN.

Examples To enable MAC-Authentication on interface `port1.0.2` and enable spanning tree edgeport to avoid unnecessary re-authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-mac enable
awplus(config-if)# spanning-tree edgeport
awplus(config-if)# switchport mode access
```

To disable MAC-Authentication on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-mac enable
```

Validation Commands
show auth-mac
show auth-mac interface
show running-config

Related Commands
aaa accounting auth-mac default
aaa authentication auth-mac
spanning-tree edgeport (RSTP and MSTP)
switchport mode access

auth-mac method

This command sets the type of authentication method for MAC-Authentication that is used with RADIUS on the interface specified in the Interface command mode.

The **no** variant of this command resets the authentication method used to the default method (PAP) as the RADIUS authentication method used by the MAC-Authentication.

Syntax

```
auth-mac method [eap-md5|pap]
no auth-mac method
```

Parameter	Description
`eap-md5`	Enable EAP-MD5 of authentication method.
`pap`	Enable PAP of authentication method.

Default The MAC-Authentication method is PAP.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the MAC-Authentication method to `pap` on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-mac method pap
```

To set the MAC-Authentication method to the default on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-mac method
```

Validation Commands
show auth-mac
show auth-mac interface
show running-config

auth-mac password

This command changes the password for MAC-based authentication.

Use the **no** variant of this command to return the password to its default.

Syntax

```
auth-mac [encrypted] password <password>
no auth-mac password
```

Parameter	Description
`auth-mac`	MAC-based authentication
`encrypted`	Specify an encrypted password
`password`	Configure the password
<password>	The new password. Passwords can be up to 64 characters in length and can contain any printable characters except ■ ? ■ " (double quotes) ■ space

Default By default, the password is the MAC address of the supplicant

Mode Global Configuration

Usage Changing the password increases the security of MAC-based authentication, because the default password is easy for an attacker to discover. This is particularly important if:

- some MAC-based supplicants on the network are intelligent devices, such as computers, and/or
- you are using two-step authentication (see **"Ensuring Authentication Methods Require Different Usernames and Passwords" on page 64.26**)

Examples To change the password to `verySecurePassword`, use the commands:

```
awplus# configure terminal
awplus(config)# auth-mac password verySecurePassword
```

Validation Command **show running-config**

Related Commands **auth two-step enable**
show auth-mac

auth-mac reauth-relearning

This command sets the MAC address learning of the supplicant (client device) to re-learning for re-authentication on the interface specified in the Interface command mode.

Use the **no** variant of this command to disable the auth-mac re-learning option.

Syntax

```
auth-mac reauth-relearning
no auth-mac reauth-relearning
```

Default Re-learning for port authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To enable the re-authentication re-learning feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-mac reauth-relearning
```

To disable the re-authentication re-learning feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-mac reauth-relearning
```

Validation Commands

show auth-mac
show auth-mac interface
show running-config

auth-web enable

This command enables Web-based authentication in Interface mode on the interface specified.

Use the **no** variant of this command to disable Web-based authentication on an interface.

Syntax

```
auth-web enable
no auth-web enable
```

Default Web-Authentication is disabled by default.

Mode Interface Configuration for a static channel or a switch port.

Usage Web-based authentication cannot be enabled if DHCP snooping is enabled (**service dhcp-snooping** command on page 80.23), and vice versa. You need to configure an IPv4 address for the VLAN interface on which Web Authentication is running.

Examples To enable Web-Authentication on static-channel-group 2, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
awplus(config)# interface sa2
awplus(config-if)# auth-web enable
```

To disable Web-Authentication on static-channel-group 2, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# static-channel-group 2
awplus(config-if)# exit
awplus(config)# interface sa2
awplus(config-if)# no auth-web enable
```

Validation Commands
show auth-web
show auth-web interface
show running-config

Related Commands
aaa accounting auth-web default
aaa authentication auth-web

auth-web forward

This command enables the Web-Authentication packet forwarding feature on the interface specified. This command also enables ARP forwarding, and adds forwarded packets to the **tcp** or **udp** port number specified.

The **no** variant of this command disables or deletes the packet forwarding feature on the interface.

Syntax

```
auth-web forward [<ip-address>] {arp|dhcp|dns|tcp <1-65535>|udp <1-65535>}
no auth-web forward [<ip-address>] {arp|dhcp|dns|tcp <1-65535>|udp <1-65535>}
```

Parameter	Description
<ip-address>	Enable forwarding to the destination IPv4 address.
arp	Enable forwarding of ARP.
dhcp	Enable forwarding of DHCP (67/udp).
dns	Enable forwarding of DNS (53/udp).
tcp	Enable forwarding of TCP specified port number.
<1-65535>	TCP Port number.
udp	Enable forwarding of UDP specified port number.
<1-65535>	UDP Port number.

Default Packet forwarding for port authentication is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage For more information about the *<ip-address>* parameter, and an example, see the "auth-web forward" section in the **Alliedware Plus Technical Tips and Tricks.**

Examples To enable the ARP forwarding feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web forward arp
```

To add the TCP forwarding port 137 on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web forward tcp 137
```

To add the DNS Server IP address 192.168.1.10 on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# switchport mode access
awplus(config-if)# auth-web enable
awplus(config-if)# auth dynamic-vlan-creation
awplus(config-if)# auth-web forward 192.168.1.10 dns
```

To disable the ARP forwarding feature on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web forward arp
```

To delete the TCP forwarding port 137 on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web forward tcp 137
```

To delete the all of TCP forwarding on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web forward tcp
```

Validation Commands

show auth-web
show auth-web interface
show running-config

auth-web max-auth-fail

This command sets the number of authentication failures allowed before rejecting further authentication requests. When the supplicant (client device) fails more than has been set to the maximum number of authentication failures then login requests are refused during the quiet period.

The **no** variant of this command resets the maximum number of authentication failures to the default (three authentication failures).

Syntax

```
auth-web max-auth-fail <0-10>
no auth-web max-auth-fail
```

Parameter	Description
<0-10>	Lock count specified.

Default The **max-auth-fail** lock counter is set to three authentication failures by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the lock count to `5` on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web max-auth-fail 5
```

To set the lock count to the default on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no auth-web max-auth-fail
```

Validation Commands **show auth-web**
show auth-web interface
show running-config

Related Commands **auth timeout quiet-period**

auth-web method

This command sets the authentication method of Web-Authentication that is used with RADIUS on the interface specified.

The **no** variant of this command sets the authentication method to PAP for the interface specified when Web-Authentication is also used with the RADIUS authentication method.

Syntax

```
auth-web method {eap-md5|pap}
no auth-web method
```

Parameter	Description
`eap-md5`	Enable EAP-MD5 as the authentication method.
`pap`	Enable PAP as the authentication method.

Default The Web-Authentication method is set to PAP by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Example To set the Web-Authentication method to eap-md5 on interface `port1.0.2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# auth-web method eap-md5
```

Validation Commands

show auth-web
show auth-web interface
show running-config

auth-web-server blocking-mode

Use this command to enable blocking mode for the Web-Authentication server. The blocking mode displays an authentication success or failure screen immediately from the response result from a RADIUS server.

Use the **no** variant of this command to disable blocking mode for the Web-Authentication server.

Syntax

```
auth-web-server blocking-mode

no auth-web-server blocking-mode
```

Parameter	Description
`blocking-mode`	Use blocking authentication server process.
`no`	Disable blocking mode.

Default By default, blocking mode is disabled for the Web-Authentication server.

Mode Global Configuration

Example To enable blocking mode for the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server blocking-mode
```

To disable blocking mode for the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server blocking-mode
```

Validation Commands **show running-config**

Related Commands **show auth-web-server**
auth-web-server mode (deleted)
auth-web-server redirect-delay-time

auth-web-server dhcp ipaddress

Use this command to assign an IP address and enable the DHCP service on the Web-Authentication server for supplicants (client devices).

Use the **no** variant of this command to remove an IP address and disable the DHCP service on the Web-Authentication server for supplicants.

Syntax

```
auth-web-server dhcp ipaddress <ip-address/prefix-length>
no auth-web-server dhcp ipaddress
```

Parameter	Description
<ip-addr/prefix-length>	The IPv4 address and prefix length assigned for the DHCP service on the Web-Authentication server for supplicants.

Default No IP address for the Web-Authentication server is set by default.

Mode Global Configuration

Usage See the section **"DHCP server for Web-Authentication" on page 64.16** in **Chapter 64, Authentication Introduction and Configuration** for further overview information about the Web-Authentication enhancements, allowing Web-Authentication to work as seamlessly as 802.1X-Authentication.

See the section **"Limitations on Allowed Feature Combinations" on page 64.33** for information about restrictions regarding combinations of authentication enhancements working together.

Examples To assign the IP address `10.0.0.1` to the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server dhcp ipaddress 10.0.0.1/8
```

To remove an IP address on the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server dhcp ipaddress
```

Validation Commands **show running-config**

Related Commands **show auth-web-server**
auth-web-server dhcp lease

auth-web-server dhcp lease

Use this command to set the DHCP lease time for supplicants (client devices) using the DHCP service on the Web-Authentication server.

Use the **no** variant of this command to reset to the default DHCP lease time for supplicants using the DHCP service on the Web-Authentication server.

Syntax

```
auth-web-server dhcp lease <20-60>
no auth-web-server dhcp lease
```

Parameter	Description
<20-60>	DHCP lease time for supplicants using the DHCP service on the Web-Authentication server in seconds.

Default The default DHCP lease time for supplicants using the DHCP service on the Web-Authentication server is set to 30 seconds.

Mode Global Configuration

Usage See the section **"DHCP server for Web-Authentication" on page 64.16** in **Chapter 64, Authentication Introduction and Configuration** for further overview information about the Web-Authentication enhancements, allowing Web-Authentication to work as seamlessly as 802.1X-Authentication.

See the section **"Limitations on Allowed Feature Combinations" on page 64.33** for information about restrictions regarding combinations of authentication enhancements working together.

Examples To set the DHCP lease time to 1 minute for supplicants using the DHCP service on the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server dhcp lease 60
```

To reset the DHCP lease time to the default setting (30 seconds) for supplicants using the DHCP service on the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server dhcp lease
```

Validation Commands **show running-config**

Related Commands **show auth-web-server**
auth-web-server dhcp ipaddress

auth-web-server dhcp-wpad-option

This command sets the DHCP WPAD (Web Proxy Auto-Discovery) option for the Web-Authentication temporary DHCP service.

For more information and examples, see the "Web Auth Proxy" section in the **Alliedware Plus Technical Tips and Tricks.**

Use the **no** variant of this command to disable the DHCP WPAD function.

Syntax

```
auth-web-server dhcp wpad-option <url>
no auth-web-server dhcp wpad-option
```

Parameter	Description
<url>	URL to the server which gets a .pac file.

Default The Web-Authentication server DHCP WPAD option is not set.

Mode Global Configuration

Usage If the supplicant is configured to use WPAD, the supplicant's web browser will use TCP port 80 as usual. Therefore, the packet can be intercepted by Web-Authentication as normal, and the Web-Authentication Login page can be sent. However, after authentication, the browser does not know where to get the WPAD file and so cannot access external web pages. The WPAD file is usually named proxy.pac file and tells the browser what web proxy to use.

Use this command to tell the supplicant where it can get this file from. The switch itself can be specified as the source for this file, and it can deliver it to the supplicant on request.

Example To specify that the proxy.pac file is found on the server at 192.168.1.100, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server dhcp wpad-option
http://192.168.1.100/proxy/
proxy.pac
```

Related Commands **show auth-web-server**

auth-web-server gateway (deleted)

This command has been deleted.

auth-web-server host-name

This command assigns a hostname to the web authentication server.

Use the **no** variant of this command to remove the hostname from the web authentication server.

Syntax

```
auth-web-server host-name <hostname>
no auth-web-server host-name
```

Parameter	Description
<hostname>	URL string of the hostname

Default The web authentication server has no hostname.

Mode Global Configuration

Usage When the web authentication server uses HTTPS protocol, the web browser will validate the certificate. If the certificate is invalid, the web page gives a warning message before displaying server content. However, the web page will not give warning message if the server has a hostname same as the one stored in the installed certificate.

Examples To set the auth.example.com as the hostname of the web authentication server, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server host-name auth.example.com
```

To remove hostname auth.example.com from the web authentication server, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server host-name
```

Related Commands **aaa authentication auth-web**
auth-web enable

auth-web-server http-redirect (deleted)

This command has been deleted.

auth-web-server intercept-port

This command specifies any additional TCP port numbers that the Web-Authentication server is to intercept.

Use the **no** variant of this command to stop intercepting the TCP port numbers.

Syntax

```
auth-web-server intercept-port <1-65535>
no auth-web-server intercept-port <1-65535>
```

Parameter	Description
<1-65535>	TCP port number.

Default No additional TCP port numbers are intercepted by default.

Mode Global Configuration

Usage If this command is not specified, AlliedWare Plus Web-Authentication intercepts the supplicant's initial TCP port 80 connection to a web page and sends it the Web-Authentication Login page. However, if the supplicant is configured to use a web proxy, then it will usually be using TCP port 8080 (or another user configured port number). In this case Web-Authentication cannot intercept the connection.

To overcome this limitation you can now use this command to tell the switch which additional port it should intercept, and then send the Web-Authentication Login page to the supplicant.

When you use this command in conjunction with a proxy server configured in the web browser, you must add the proxy server's network as a 'No Proxy' network. You can specify 'No Proxy' networks in the proxy settings in your web browser. For more information, see the "Web Auth Proxy" section in the **Alliedware Plus Technical Tips and Tricks.**

Example To additionally intercept port number 3128, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server intercept-port 3128
```

Related Commands **show auth-web-server**

auth-web-server ipaddress

This command sets the IP address for the Web-Authentication server.

Use the **no** variant of this command to delete the IP address for the Web-Authentication server.

Syntax

```
auth-web-server ipaddress <ip-address>
no auth-web-server ipaddress
```

Parameter	Description
<ip-address>	Web-Authentication server dotted decimal IP address in A.B.C.D format.

Default The Web-Authentication server address on the system is not set by default.

Mode Global Configuration

Examples To set the IP address `10.0.0.1` to the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ipaddress 10.0.0.1
```

To delete the IP address from the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ipaddress
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server login-url

This command sets the web-authentication login page URL.

Use the **no** variant of this command to delete the set URL.

Syntax

```
auth-web-server login-url <URL>
no auth-web-server login-url
```

Parameter	Description
<URL>	Set login page URL

Default The built-in login page is set by default.

Mode Global Configuration

Examples To set http://example.com/login.html as the login page, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server login-url http://
example.com/login.html
```

To unset the login page URL, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server login-url
```

Validation Commands **show running-config**

auth-web-server mode (deleted)

This command has been deleted.

auth-web-server page logo

This command sets the type of logo that will be displayed on the web authentication page.

Use the **no** variant of this command to set the logo type to **auto**.

Syntax

```
auth-web-server page logo {auto|default|hidden}
no auth-web-server page logo
```

Parameter	Description
`auto`	Display the custom logo if installed; otherwise display the default logo
`default`	Display the default logo
`hidden`	Hide the logo

Default Logo type is **auto** by default.

Mode Global Configuration

Examples To display the default logo with ignoring installed custom logo, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page logo default
```

To set back to the default logo type **auto**, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server page logo
```

Validation Commands **show auth-web-server page**

auth-web-server page sub-title

This command sets the custom sub-title on the web authentication page.

Use the **no** variant of this command to reset the sub-title to its default.

Syntax

```
auth-web-server page sub-title {hidden|text <sub-title>}
no auth-web-server page sub-title
```

Parameter	Description
`hidden`	Hide the sub-title
<sub-title>	Text string of the sub-title

Default "Allied-Telesis" is displayed by default.

Mode Global Configuration

Examples To set the custom sub-title, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page sub-title text Web
Authentication
```

To hide the sub-title, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page sub-title hidden
```

To change back to the default title, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server page sub-title
```

Validation Commands **show auth-web-server page**

auth-web-server page success-message

This command sets the success message on the web-authentication page.

Use the **no** variant of this command to remove the success message.

Syntax

```
auth-web-server page success-message text <success-message>
no auth-web-server page success-message
```

Parameter	Description
<success-message>	Text string of the success message

Default No success message is set by default.

Mode Global Configuration

Examples To set the success message on the web-authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page success-message text
Your success message
```

To unset the success message on the web-authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server page success-message
```

Validation Commands **show auth-web-server page**

auth-web-server page title

This command sets the custom title on the web authentication page.

Use the **no** variant of this command to remove the custom title.

Syntax

```
auth-web-server page title {hidden|text <title>}
no auth-web-server page title
```

Parameter	Description
`hidden`	Hide the title
`<title>`	Text string of the title

Default "Web Access Authentication Gateway" is displayed by default.

Mode Global Configuration

Examples To set the custom title on the web authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page title text Login
```

To hide the title on the web authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page title hidden
```

To unset the custom title on the web authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server page title
```

Validation Commands **show auth-web-server page**

auth-web-server page welcome-message

This command sets the welcome message on the web-authentication page.

Use the **no** variant of this command to remove the welcome message.

Syntax

```
auth-web-server page welcome-message text <welcome-message>
no auth-web-server page welcome-message
```

Parameter	Description
<welcome-message>	Text string of the welcome message

Default No welcome message is set by default.

Mode Global Configuration

Examples To set the welcome message on the web-authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server page welcome-message text
Your welcome message
```

To remove the welcome message on the web-authentication page, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server page welcome-message
```

Validation Commands **show auth-web-server page**

auth-web-server ping-poll enable

This command enables the ping polling to the supplicant (client device) that is authenticated by Web-Authentication.

The **no** variant of this command disables the ping polling to the supplicant that is authenticated by Web-Authentication.

Syntax

```
auth-web-server ping-poll enable
no auth-web-server ping-poll enable
```

Default The ping polling feature for Web-Authentication is disabled by default.

Mode Global Configuration

Examples To enable the ping polling feature for Web-Authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ping-poll enable
```

To disable the ping polling feature for Web-Authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ping-poll enable
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll failcount

This command sets a fail count for the ping polling feature when used with Web-Authentication. The **failcount** parameter specifies the number of unanswered pings. A supplicant (client device) is logged off when the number of unanswered pings are greater than the failcount set with this command.

Use the **no** variant of this command to resets the fail count for the ping polling feature to the default (5 pings).

Syntax

```
auth-web-server ping-poll failcount <1-100>
no auth-web-server ping-poll failcount
```

Parameter	Description
<1-100>	Count.

Default The default failcount for ping polling is 5 pings.

Mode Global Configuration

Examples To set the failcount of ping polling to `10` pings, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ping-poll failcount 10
```

To set the failcount of ping polling to default, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ping-poll failcount
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll interval

This command is used to change the ping poll interval. The interval specifies the time period between pings when the supplicant (client device) is reachable.

Use the **no** variant of this command to reset to the default period for ping polling (30 seconds).

Syntax

```
auth-web-server ping-poll interval <1-65535>
no auth-web-server ping-poll interval
```

Parameter	Description
<1-65535>	Seconds.

Default The interval for ping polling is 30 seconds by default.

Mode Global Configuration

Examples To set the interval of ping polling to `60` seconds, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ping-poll interval 60
```

To set the interval of ping polling to the default (30 seconds), use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ping-poll interval
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll reauth-timer-refresh

This command modifies the **reauth-timer-refresh** parameter for the Web-Authentication feature. The **reauth-timer-refresh** parameter specifies whether a re-authentication timer is reset and when the response from a supplicant (a client device) is received.

Use the **no** variant of this command to reset the **reauth-timer-refresh** parameter to the default setting (disabled).

Syntax

```
auth-web-server ping-poll reauth-timer-refresh
no auth-web-server ping-poll reauth-timer-refresh
```

Default The `reauth-timer-refresh` parameter is disabled by default.

Mode Global Configuration

Examples To enable the `reauth-timer-refresh` timer, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ping-poll reauth-timer-refresh
```

To disable the `reauth-timer-refresh` timer, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ping-poll reauth-timer-
refresh
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ping-poll timeout

This command modifies the ping poll **timeout** parameter for the Web-Authentication feature. The **timeout** parameter specifies the time in seconds to wait for a response to a ping packet.

Use the **no** variant of this command to reset the timeout of ping polling to the default (1 second).

Syntax

```
auth-web-server ping-poll timeout <1-30>
no auth-web-server ping-poll timeout
```

Parameter	Description
<1-30>	Seconds.

Default The default timeout for ping polling is 1 second.

Mode Global Configuration

Examples To set the timeout of ping polling to 2 seconds, use the command:

```
awplus# configure terminal
awplus(config)# auth-web-server ping-poll timeout 2
```

To set the timeout of ping polling to the default (1 second), use the command:

```
awplus# configure terminal
awplus(config)# no auth-web-server ping-poll timeout
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server port

This command sets the HTTP port number for the Web-Authentication server.

Use the **no** variant of this command to reset the HTTP port number to the default (80).

Syntax

```
auth-web-server port <port-number>
no auth-web-server port
```

Parameter	Description
<port-number>	Set the local Web-Authentication server port within the TCP port number range 1 to 65535.

Default The Web-Authentication server HTTP port number is set to 80 by default.

Mode Global Configuration

Examples To set the HTTP port number `8080` for the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server port 8080
```

To reset to the default HTTP port number `80` for the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server port
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server redirect-delay-time

Use this command to set the delay time in seconds before redirecting the supplicant to a specified URL when the supplicant is authorized.

Use the variant **no** to reset the delay time set previously.

Syntax

```
auth-web-server redirect-delay-time <5-60>
no auth-web-server redirect-delay-time
```

Parameter	Description
`redirect-delay-time`	Set the delay time before jumping to a specified URL after the supplicant is authorized.
<5-60>	The time in seconds.

Default The default redirect delay time is `5` seconds.

Mode Global Configuration

Examples To set the delay time to `60` seconds for the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server redirect-delay-time 60
```

To reset the delay time, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server redirect-delay-time
```

Validation Command **show auth-web-server**
show running-config

Related Commands **auth-web-server redirect-url**
show auth-web-server

auth-web-server redirect-url

This command sets a URL for supplicant (client device) authentication. When a supplicant is authorized it will be automatically redirected to the specified URL. Note that if the http redirect feature is used then this command is ignored.

Use the **no** variant of this command to delete the URL string set previously.

Syntax

```
auth-web-server redirect-url <url>
no auth-web-server redirect-url
```

Parameter	Description
<url>	URL (hostname or dotted IP notation).

Default The redirect URL for the Web-Authentication server feature is not set by default (null).

Mode Global Configuration

Examples To enable and set redirect a URL string `www.alliedtelesis.com` for the Web-Authentication server, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server redirect-url
http://www.alliedtelesis.com
```

To delete a redirect URL string, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server redirect-url
```

Validation Commands
show auth-web
show auth-web-server
show running-config

Related Commands
auth-web-server http-redirect (deleted)
auth-web-server redirect-delay-time

auth-web-server session-keep

This command enables the session-keep feature to jump to the original URL after being authorized by Web-Authentication.

Use the **no** variant of this command to disable the session keep feature.

Syntax

```
auth-web-server session-keep
no auth-web-server session-keep
```

Default The session-keep feature is disabled by default.

Mode Global Configuration

Examples To enable the session-keep feature, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server session-keep
```

To disable the session-keep feature, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server session-keep
```

Validation Commands

show auth-web
show auth-web-server
show running-config

auth-web-server ssl

This command enables HTTPS functionality for the Web-Authentication server feature.

Use the **no** variant of this command to disable HTTPS functionality for the Web-Authentication server.

Syntax

```
auth-web-server ssl
no auth-web-server ssl
```

Default HTTPS functionality for the Web-Authentication server feature is disabled by default.

Mode Global Configuration

Examples To enable HTTPS functionality for the Web-Authentication server feature, use the following commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ssl
```

To disable HTTPS functionality for the Web-Authentication server feature, use the following commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ssl
```

Validation Commands
show auth-web
show auth-web-server
show running-config

auth-web-server sslport (deleted)

This command has been deleted.

auth-web-server ssl intercept-port

Use this command to register HTTPS intercept port numbers when the HTTPS server uses custom port number (not TCP port number 443).

Note that you need to use the **auth-web-server intercept-port** command to register HTTP intercept port numbers.

Use the **no** variant of this command to delete registered port number.

Syntax

```
auth-web-server ssl intercept-port <1-65535>
no auth-web-server ssl intercept-port <1-65535>
```

Parameter	Description
<1-65535>	TCP port number in the range from 1 through 65535

Default 443/TCP is registered by default.

Mode Global Configuration

Examples To register HTTPS port number 3128, use the commands:

```
awplus# configure terminal
awplus(config)# auth-web-server ssl intercept-port 3128
```

To delete HTTPS port number 3128, use the commands:

```
awplus# configure terminal
awplus(config)# no auth-web-server ssl intercept-port 3128
```

Validation Commands **show auth-web-server**

Related Commands **auth-web-server intercept-port**

copy proxy-autoconfig-file

Use this command to download the proxy auto configuration (PAC) file to your switch. The Web-Authentication supplicant can get the downloaded file from the system web server.

Syntax `copy <filename> proxy-autoconfig-file`

Parameter	Description
<filename>	The URL of the PAC file.

Mode Privileged Exec

Example To download the PAC file to this device, use the command:

```
awplus# copy tftp://server/proxy.pac proxy-autoconfig-file
```

Related Commands **show proxy-autoconfig-file**
erase proxy-autoconfig-file

copy web-auth-https-file

Use this command to download the SSL server certificate for web-based authentication. The file must be in PEM (Privacy Enhanced Mail) format, and contain the private key and the server certificate.

Syntax `copy <filename> web-auth-https-file`

Parameter	Description
<filename>	The URL of the server certificate file.

Mode Privileged Exec

Example To download the server certificate file `veriSign_cert.pem` from the TFTP server directory `server`, use the command:

```
awplus# copy tftp://server/veriSign_cert.pem web-auth-https-
       file
```

Related Commands **auth-web-server ssl**
erase web-auth-https-file
show auth-web-server

erase proxy-autoconfig-file

Use this command to remove the proxy auto configuration file.

Syntax `erase proxy-autoconfig-file`

Mode Privileged Exec

Example To remove the proxy auto configuration file, use the command:

```
awplus# erase proxy-autoconfig-file
```

Related Commands **show proxy-autoconfig-file**
copy proxy-autoconfig-file

erase web-auth-https-file

Use this command to remove the SSL server certificate for web-based authentication.

Syntax `erase web-auth-https-file`

Mode Privileged Exec

Example To remove the SSL server certificate file for web-based authentication use the command:

```
awplus# erase web-auth-https-file
```

Related Commands **auth-web-server ssl**
copy web-auth-https-file
show auth-web-server

platform l3-vlan-hashing-algorithm

This command enables you to change the L3 VLAN hash-key-generating algorithm.

The **no** variant of this command returns the hash-key algorithm to the default of `crc32l`.

Syntax

```
platform l3-vlan-hashing-algorithm {crc16l|crc16u|crc32l|crc32u}
no platform l3-vlan-hashing-algorithm
```

Parameter	Description
`crc16l`	The algorithm that will apply to the lower bits of CRC-16
`crc16u`	The algorithm that will apply to the upper bits of CRC-16
`crc32l`	The algorithm that will apply to the lower bits of CRC-32
`crc32u`	The algorithm that will apply to the upper bits of CRC-32

Default The hash-key algorithm is `crc32l` by default.

Mode Global configuration

Usage Occasionally, when using the Multiple Dynamic VLAN feature, a supplicant cannot be authenticated because a collision occurs within the VLAN L3 table. This can happen when more than four different IP addresses produce the same hash-key.

A work-around when this situation occurs, can sometimes be applied by changing the hashing algorithm from its default of `crc32l`. Several different algorithms may need to be tried to rectify the problem.

You must restart the switch for this command to take effect.

Note that this command is intended for technical support staff, or advanced end users.

Example To change the hash-key generating algorithm applying to the lower bits of CRC-16, use the command:

```
awplus# configure terminal
awplus(config)# platform l3-vlan-hashing-algorithm crc16l
```

Related Commands

platform mac-vlan-hashing-algorithm
show platform

platform mac-vlan-hashing-algorithm

This command enables you to change the MAC VLAN hash-key-generating algorithm.

The **no** variant of this command returns the hash-key algorithm to the default of `crc32l`.

Syntax

```
platform mac-vlan-hashing-algorithm {crc16l|crc16u|crc32l|crc32u}
no platform mac-vlan-hashing-algorithm
```

Parameter	Description
`crc16l`	The algorithm that will apply to the lower bits of CRC-16
`crc16u`	The algorithm that will apply to the upper bits of CRC-16
`crc32l`	The algorithm that will apply to the lower bits of CRC-32
`crc32u`	The algorithm that will apply to the upper bits of CRC-32

Default The hash-key algorithm is `crc32l` by default.

Mode Global configuration

Usage Occasionally, when using the Multiple Dynamic VLAN feature, a supplicant cannot be authenticated because a collision occurs within the VLAN MAC table. This can happen when more than four different MAC addresses produce the same hash-key.

A work-around when this situation occurs, can sometimes be applied by changing the hashing algorithm from its default of `crc32l`. Several different algorithms may need to be tried to rectify the problem.

You must restart the switch for this command to take effect.

Note that this command is intended for technical support staff, or advanced end users.

Example To change the hash-key generating algorithm applying to the lower bits of CRC-16, use the command:

```
awplus# configure terminal
awplus(config)# platform mac-vlan-hashing-algorithm crc16l
```

Related Commands

platform l3-vlan-hashing-algorithm
show platform

show auth two-step supplicant brief

This command displays the supplicant state of the two-step authentication feature on the interface.

Syntax `show auth two-step supplicant [interface <ifrange>] brief`

Parameter	Description
`interface`	The interface selected for display.
<ifrange>	The interface which can be specified as *<ifrange>* - Switch port (e.g. port1.0.12) - Static channel group (e.g. sa3) - Dynamic (LACP) channel group (e.g. po4)

Mode Privileged Exec

Usage Do not mix interface types in a list. The specified interfaces must exist.

Example To display the supplicant state of the two-step authentication feature, enter the command:

```
awplus# show two-step supplicant interface
       port1.0.12 brief
```

Output **Figure 65-1: Example output from the show auth two-step supplicant brief command**

```
interface port1.0.12
  authenticationMethod: dot1x/mac
  Two-Step Authentication:
    firstMethod:mac
    secondMethod:dot1x
  totalSupplicantNum: 1
  authorizedSupplicantNum: 1
    macBasedAuthenticationSupplicantNum: 0
    dot1xAuthenticationSupplicantNum: 1
    webBasedAuthenticationSupplicantNum: 0
    otherAuthenticationSupplicantNum: 0

Interface  VID Mode  MAC Address       Status          FirstStep     SecondStep
========== === ====  ==========        ======          =========     ==========
port1.0.12   1   D     000b..db67.00f7   Authenticated   Pass          Pass
```

Related Commands **auth two-step enable**

show auth-mac

This command shows authentication information for MAC-based authentication.

Syntax `show auth-mac [all]`

Parameter	Description
`all`	Display all authentication information for each interface available on the switch.

Mode Privileged Exec

Example To display all MAC-based authentication information, enter the command:

```
awplus# show auth-mac all
```

Output **Figure 65-2: Example output from the show auth-mac command**

```
802.1X Port-Based Authentication Disabled
MAC-based Port Authentication Enabled
WEB-based Port Authentication Disabled
```

Related Commands **show dot1x**
show auth-web

show auth-mac diagnostics

This command shows MAC-Authentication diagnostics, optionally for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

If no interface is specified then authentication diagnostics are shown for all interfaces.

Syntax `show auth-mac diagnostics [interface <interface-list>]`

Parameter	Description
`interface`	Specify an interface to show
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display authentication diagnostics for `port1.0.6`, enter the command:

```
awplus# show auth-mac diagnostics interface port1.0.6
```

Output **Figure 65-3: Example output from the show auth-mac diagnostics command**

```
Authentication Diagnostics for interface port1.0.6
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

show auth-mac interface

This command shows the status for MAC-based authentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the optional **diagnostics** parameter to show authentication diagnostics for the specified interface. Use the optional **sessionstatistics** parameter to show authentication session statistics for the specified interface. Use the optional **statistics** parameter to show authentication diagnostics for the specified interface. Use the optional **supplicant** (client device) parameter to show the supplicant state for the specified interface.

Syntax

```
show auth-mac interface <interface-list> [diagnostics|
   sessionstatistics|statistics|supplicant [brief]]
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`diagnostics`	Diagnostics.
`sessionstatistics`	Session statistics.
`statistics`	Statistics.
`supplicant`	Supplicant (client device).
`brief`	Brief summary of supplicant state.

Mode Privileged Exec

Examples To display MAC-based authentication status for `port1.0.2`, enter the command:

```
awplus# show auth-mac interface port1.0.2
```

```
% Port-Control not configured on port1.0.2
```

To display MAC-Authentication diagnostics for `port1.0.2`, enter the command:

```
awplus# show auth-mac interface port1.0.2 diagnostics
```

```
Authentication Diagnostics for interface port1.0.2
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

To display authentication session statistics for `port1.0.6`, enter the command:

```
awplus# show auth-mac interface port1.0.6 sessionstatistics
```

```
Authentication session statistics for interface port1.0.6
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

To display MAC-Authentication statistics for `port1.0.6` enter the command:

```
awplus# show auth-mac interface port1.0.6 statistics
```

To display the MAC authenticated supplicant on interface `port1.0.6`, enter the command:

```
awplus# show auth-mac interface port1.0.6 supplicant
```

Related Commands

show auth-web diagnostics
show dot1x sessionstatistics
show dot1x statistics interface
show dot1x supplicant interface

show auth-mac sessionstatistics

This command shows authentication session statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `show auth-mac sessionstatistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify an interface to show.
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display output displaying MAC-Authentication session statistics for `port1.0.2`, enter the command:

```
awplus# show auth-mac sessionstatistics interface port1.0.2
```

Output **Figure 65-4: Example output from the show auth-mac sessionstatistics command**

```
Authentication session statistics for interface port1.0.2
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

show auth-mac statistics interface

This command shows the authentication statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `show auth-mac statistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify ports to show.
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display MAC-Authentication statistics for `port1.0.2`, enter the command:

```
awplus# show auth-mac statistics interface port1.0.2
```

Related Commands **show dot1x interface**

show auth-mac supplicant

This command shows the supplicant (client device) state when MAC-Authentication is configured for the switch. This command shows a summary when the optional **brief** parameter is used.

Syntax `show auth-mac supplicant [<macadd>] [brief]`

Parameter	Description
<macadd>	Mac (hardware) address of the Supplicant. Entry format is HHHH.HHHH.HHHH (hexadecimal).
`brief`	Brief summary of the Supplicant state.

Mode Privileged Exec

Example To display the MAC authenticated supplicant for MAC address `00d0.59ab.7037`, enter the command:

```
awplus# show auth-mac supplicant 00d0.59ab.7037
```

```
Web authentication server
    Server status: enabled
    Server address: -
    HTTP Port No: 80
    Security: enabled
    Certification: default
    SSL Port No: 443
    Redirect URL:
    Redirect Delay Time: 30
    HTTP Redirect: disabled
    Session keep: disabled
    PingPolling: disable
    PingInterval: 30
    Timeout: 1
    FailCount: 5
    ReauthFresh: disabled
```

Example To display a brief summary output for a MAC authenticated supplicant, enter the command:

```
awplus# show auth-mac supplicant brief
```

For example, if two-step authentication is configured with MAC-Authentication as the first method and 802.1X-Authentication as the second method then the output is as follows:

```
Interface port1.0.6
  authenticationMethod: dot1x/mac
  Two-Step Authentication
    firstMethod: mac
    secondMethod: dot1x
  totalSupplicantNum: 1
  authorizedSupplicantNum: 1
    macBasedAuthenticationSupplicantNum: 0
    dot1xAuthenticationSupplicantNum: 1
    webBasedAuthenticationSupplicantNum: 0
    otherAuthenticationSupplicantNum: 0

Interface   VID  Mode MAC Address    Status            IP Address      Username
=========== ==== ==== ============== ================= ==============  =========
port1.0.6   5    D    0008.0d5e.c216 Authenticated     --              dot1x
```

For example, if two-step authentication is configured with MAC-Authentication as the first method and Web-Authentication as the second method then the output is as follows:

```
Interface port1.0.7
  authenticationMethod: mac/web
  Two-Step Authentication
    firstMethod: mac
    secondMethod: web
  totalSupplicantNum: 1
  authorizedSupplicantNum: 1
    macBasedAuthenticationSupplicantNum: 0
    dot1xAuthenticationSupplicantNum: 0
    webBasedAuthenticationSupplicantNum: 1
    otherAuthenticationSupplicantNum: 0

Interface   VID  Mode MAC Address    Status            IP Address      Username
=========== ==== ==== ============== ================= =============== =========
port1.0.7   5    W    0008.0d5e.c216 Authenticated     192.168.1.200   web
```

show auth-mac supplicant interface

This command shows the supplicant (client device) state for the MAC authenticated interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port. This command shows a summary when the optional **brief** parameter is used.

Syntax `show auth-mac supplicant [interface <interface-list>] [brief]`

Parameter	Description
`interface`	Specify ports to show.
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`brief`	Brief summary of the supplicant state.

Mode Privileged Exec

Examples To display the MAC authenticated supplicant on the interface `port1.0.2`, enter the command:

```
awplus# show auth-mac supplicant interface port1.0.2
```

show auth-web

This command shows authentication information for Web-based authentication.

Syntax `show auth-web [all]`

Parameter	Description
`all`	Display all authentication information for each authenticated interface. This can be a static channel (or static aggregator), or a dynamic (or LACP) channel group, or a switch port.

Mode Privileged Exec

Example To display all Web-Authentication information, enter the command:

```
awplus# show auth-web all
```

Output **Figure 65-5: Example output from the show auth-web command**

```
awplus# show auth-web all
802.1X Port-Based Authentication Enabled
MAC-based Port Authentication Disabled
WEB-based Port Authentication Enabled
 RADIUS server address (auth): 150.87.17.192:1812
   Last radius message id: 4
Authentication Info for interface port1.0.1
 portEnabled: true - portControl: Auto
 portStatus: Authorized
 reAuthenticate: disabled
 reAuthPeriod: 3600
 PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
 BE: suppTimeout: 30 - serverTimeout: 30
 CD: adminControlledDirections: in
 KT: keyTxEnabled: false
 critical: disabled
 guestVlan: disabled
 authFailVlan: disabled
 dynamicVlanCreation: disabled
 hostMode: single-host
 dot1x: enabled
   protocolVersion: 1
 authMac: disabled
 authWeb: enabled
   method: PAP
   maxAuthFail: 3
   packetForwarding:
     10.0.0.1 80/tcp
     dns
     dhcp
 twoStepAuthentication:
   configured: enabled
   actual: enabled
 supplicantMac: none
 Supplicant name: oha
 Supplicant address: 000d.6013.5398
   authenticationMethod: WEB-based Authentication
   Two-Step Authentication:
     firstAuthentication: Pass - Method: dot1x
     secondAuthentication: Pass - Method: web
   portStatus: Authorized - currentId: 3
   abort:F fail:F start:F timeout:F success:T
   PAE: state: Authenticated - portMode: Auto
   PAE: reAuthCount: 0 - rxRespId: 0
   PAE: quietPeriod: 60 - maxReauthReq: 2
   BE: state: Idle - reqCount: 0 - idFromServer: 2
   CD: adminControlledDirections: in - operControlledDirections: in
   CD: bridgeDetected: false
   KR: rxKey: false
   KT: keyAvailable: false - keyTxEnabled: false
```

Related Commands **show dot1x**
show auth-mac

show auth-web diagnostics

This command shows Web-Authentication diagnostics, optionally for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

If no interface is specified then authentication diagnostics are shown for all interfaces.

Syntax `show auth-web diagnostics [interface <interface-list>]`

Parameter	Description
`interface`	Specify ports to show.
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display authentication diagnostics for `port1.0.6`, enter the command:

```
awplus# show auth-web diagnostics interface port1.0.6
```

Output **Figure 65-6: Example output from the show auth-web diagnostics command**

```
Authentication Diagnostics for interface port1.0.6
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

Related Commands **show dot1x interface**

show auth-web interface

This command shows the status for Web based authentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the optional **diagnostics** parameter to show authentication diagnostics for the specified interface. Use the optional **sessionstatistics** parameter to show authentication session statistics for the specified interface. Use the optional **statistics** parameter to show authentication diagnostics for the specified interface. Use the optional **supplicant** (client device) parameter to show the supplicant state for the specified interface.

Syntax

```
show auth-web interface <interface-list> [diagnostics|
   sessionstatistics|statistics|supplicant [brief]]
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`diagnostics`	Diagnostics.
`sessionstatistics`	Session statistics.
`statistics`	Statistics.
`supplicant`	Supplicant (client device).
`brief`	Brief summary of supplicant state.

Mode Privileged Exec

Example To display the Web based authentication status for `port1.0.6`, enter the command:

```
awplus# show auth-web interface port1.0.6
```

To display the Web based authentication status for `port1.0.1`, enter the command:

```
awplus# show auth-web interface port1.0.1
```

```
awplus# show auth-web interface port1.0.1
Authentication Info for interface port1.0.1
 portEnabled: true - portControl: Auto
 portStatus: Authorized
 reAuthenticate: disabled
 reAuthPeriod: 3600
 PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
 BE: suppTimeout: 30 - serverTimeout: 30
 CD: adminControlledDirections: in
 KT: keyTxEnabled: false
 critical: disabled
 guestVlan: disabled
 authFailVlan: disabled
 dynamicVlanCreation: disabled
 hostMode: single-host
 dot1x: enabled
   protocolVersion: 1
 authMac: disabled
 authWeb: enabled
   method: PAP
   maxAuthFail: 3
   packetForwarding:
     10.0.0.1 80/tcp
     dns
     dhcp
 twoStepAuthentication:
   configured: enabled
   actual: enabled
 supplicantMac: none
```

To display Web-Authentication diagnostics for `port1.0.6`, enter the command:

```
awplus# show auth-web interface port1.0.6 diagnostics
```

```
Authentication Diagnostics for interface port1.0.6
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

To display Web-Authentication session statistics for `port1.0.6`, enter the command:

```
awplus# show auth-web interface port1.0.6 sessionstatistics
```

```
Authentication session statistics for interface port1.0.6
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

To display Web-Authentication statistics for `port1.0.6`, enter the command:

```
awplus# show auth-web statistics interface port1.0.6
```

To display the Web-Authenticated supplicant on interface `port1.0.6`, enter the command:

```
awplus# show auth-web interface port1.0.6 supplicant
```

Related Commands

show auth-web diagnostics
show dot1x sessionstatistics
show dot1x statistics interface
show dot1x supplicant interface

show auth-web sessionstatistics

This command shows authentication session statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `show auth-web sessionstatistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify ports to show.
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display authentication statistics for `port1.0.6`, enter the command:

```
awplus# show auth-web sessionstatistics interface port1.0.6
```

Output **Figure 65-7: Example output from the show auth-web sessionstatistics command**

```
Authentication session statistics for interface port1.0.6
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

show auth-web statistics interface

This command shows the authentication statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Syntax `show auth-web statistics interface <interface-list>`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example To display Web-Authentication statistics for `port1.0.4`, enter the command:

```
awplus# show dot1x statistics interface port1.0.4
```

Related Commands **show dot1x interface**

show auth-web supplicant

This command shows the supplicant (client device) state when Web-Authentication is configured for the switch. This command shows a summary when the optional **brief** parameter is used.

Syntax `show auth-web supplicant [<macadd>] [brief]`

Parameter	Description
`<macadd>`	Mac (hardware) address of the supplicant. Entry format is HHHH.HHHH.HHHH (hexadecimal).
`brief`	Brief summary of the supplicant state.

Mode Privileged Exec

Examples To display Web authenticated supplicant information on the switch, enter the command:

```
awplus# show auth-web supplicant
```

show auth-web supplicant interface

This command shows the supplicant (client device) state for the Web authenticated interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port. This command shows a summary when the optional **brief** parameter is used.

Syntax `show auth-web supplicant interface <interface-list> [brief]`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`brief`	Brief summary of the supplicant state.

Mode Privileged Exec

Examples To display the Web authenticated supplicant on the interface `port1.0.3`, enter the command:

```
awplus# show auth-web supplicant interface port1.0.3
```

To display brief summary output for the Web authenticated supplicant, enter the command:

```
awplus# show auth-web supplicant brief
```

show auth-web-server

This command shows the Web-Authentication server configuration and status on the switch.

Syntax `show auth-web-server`

Mode Privileged Exec

Example To display Web-Authentication server configuration and status, enter the command:

```
awplus# show auth-web-server
```

Output **Figure 65-8: Example output from the show auth-web-server command**

```
Web authentication server
    Server status: enabled
    Server mode: none
    Server address: 192.168.1.1/24
      DHCP server enabled
      DHCP lease time: 20
      DHCP WPAD Option URL: http://192.168.1.1/proxy.pac
    HTTP Port No: 80
    Security: disabled
    Certification: default
    SSL Port No: 443
    Redirect URL: --
    Redirect Delay Time: 5
    HTTP Redirect: enabled
    Session keep: disabled
    PingPolling: disabled
    PingInterval: 30
    Timeout: 1
    FailCount: 5
    ReauthTimerReFresh: disabled
```

Related Commands
auth-web-server gateway (deleted)
auth-web-server http-redirect (deleted)
auth-web-server ipaddress
auth-web-server port
auth-web-server redirect-delay-time
auth-web-server redirect-url
auth-web-server session-keep
auth-web-server ssl
auth-web-server sslport (deleted)

show auth-web-server page

This command displays the web-authentication page configuration and status.

Syntax `show auth-web-server page`

Mode Privileged Exec

Examples To show the web-authentication page information, use the command:

```
awplus# show auth-web-server page
```

Figure 66: Example output from the show auth-web-server page command on the console.

```
awplus#show auth-web-server page
Web authentication page
  Logo: auto
  Title: default
  Sub-Title: Web Authentication
  Welcome message: Your welcome message
  Success message: Your success message
```

Related Commands
auth-web forward
auth-web-server page logo
auth-web-server page sub-title
auth-web-server page success-message
auth-web-server page title
auth-web-server page welcome-message

show proxy-autoconfig-file

This command displays the contents of the proxy auto configuration (PAC) file.

Syntax `show proxy-autoconfig-file`

Mode Privileged Exec

Example To display the contents of the proxy auto configuration (PAC) file, enter the command:

```
awplus# show auth proxy-autoconfig-file
```

Output **Figure 65-9: Example output from the show proxy-autoconfig-file**

```
function FindProxyForURL(url,host)
{
 if (isPlainHostName(host) ||
     isInNet(host, "192.168.1.0","255.255.255.0")) {
   return "DIRECT";
}
 else {
   return "PROXY 192.168.110.1:8080";
 }
}
```

Related Commands **copy proxy-autoconfig-file**
erase proxy-autoconfig-file

Chapter 66: AAA Introduction and Configuration

AAA Introduction 66.2
Available Functions and Server Types 66.2
Server Groups and Method Lists 66.3
Configuring AAA Login Authentication 66.5
AAA Configuration Tasks 66.5
Sample Authentication Configurations 66.7
Sample 802.1X-Authentication Configuration 66.7
Sample MAC-Authentication Configuration 66.8
Sample Web-Authentication Configuration 66.9
Sample Tri-Authentication Configuration 66.10

AAA Introduction

AAA is the collective title for the three related functions of Authentication, Authorization and Accounting. These function can be applied in a variety of methods with a variety of servers. The purpose of the AAA commands is to map instances of the AAA functions to sets of servers.

The Authentication function can be performed in multiple contexts, such as authentication of users logging in at a console, or 802.1X-Authentication of devices connecting to Ethernet ports.

For each of these contexts, you may want to use different sets of servers for examining the proffered authentication credentials and deciding if they are valid. AAA Authentication commands enable you to specify which servers will be used for different types of authentication.

Available Functions and Server Types

Authentication, Authorization and Accounting functions are available.

Authentication is performed in the following contexts:

- Login authentication of user shell sessions on the console port, and via telnet/SSH
- Enable password authentication for user shell sessions on the console port, and via telnet/SSH (TACACS+ only)
- 802.1X-Authentication of devices connecting to switch ports
- MAC-Authentication of devices connecting to switch ports
- Web-based authentication of devices connecting to switch ports

Authorization is performed in the following context:

- TACACS+ login authentication. Note that with the AlliedWare Plus TACACS+ implementation:
  - « authorization cannot be performed independently of the login authentication process
  - « authorization will not be attempted if enable password authentication is configured
  - « there are no authorization commands available

Accounting is performed in the following contexts:

- Accounting of console, telnet, and SSH login sessions
- Accounting of commands executed within user shell sessions (TACACS+ only)
- Accounting of 802.1X-Authenticated connections
- Accounting of MAC-Authenticated connections
- Accounting of Web-Authenticated connections

The three types of servers that can be used are:

- Local user database
- RADIUS servers
- TACACS+ servers

Server Groups and Method Lists

There are two constructs that underlie the structure of the AAA commands:

- Server groups are lists of RADIUS servers
- Method Lists are lists of server types

Server groups

A server group is defined by the command **aaa group server**. This command puts you into Server Group configuration mode. Once in that mode you can add servers to the group by using the command **server auth-port**.

Any number of servers can be added to a group. Typically, you will add servers which have already been configured by the command **radius-server host**. If you add a server that has not yet been configured by the command **radius-server host**, you will receive a warning that the server has not yet been configured, but the command will be accepted.

There is one server group that is always present on the switch by default that cannot be removed. It is the group simply named **radius** that comprises all servers that have been configured using the command **radius-server host**. As soon as a server is configured by the command **radius-server host**, it is automatically a member of the server group **radius** and cannot be removed from it.

Method lists

A method list defines the set of server types that you want to be used for authenticating a user/device, and the order in which you want the server types to be used.

- You may want the usernames proffered for logging in at the console to be checked for in the local user database. You can create a server list that specifies **local**.
- You may want to check the TACACS+ servers first, and resort to the local user database if none of the TACACS+ servers respond. You can create a server list that specifies **group tacacs+** first, followed by **local**.
- You may want to check the RADIUS servers first, and resort to the local user database if none of the RADIUS servers respond. You can create a server list that specifies **group radius** first, followed by **local**.

A method list defines the servers where authentication requests are sent. The first server listed is used to authenticate users; if that server fails then the next authentication server type in the method list is selected. This process continues until there is a successful authentication or until all server types fail.

When a user attempts to log in, the switch sends an authentication request to the first authentication server in the method list. If the first server in the list is reachable and it contains a username and password matching the authentication request, the user is authenticated and the login succeeds. If the authentication server denies the authentication request because of an incorrect username or password, the user login fails. If the first server in the method list is unreachable, the switch sends the request to the next server in the list, and so on.

For example, if the method list specifies `group tacacs+ local`, and a user attempts to log in with a password that does not match a user entry in the first TACACS+ server, if this TACACS+ server denies the authentication request, then the switch does not try any other TACACS+ servers not the local user database; the user login fails.

Default method lists

For every authentication or accounting type, it is always possible to define a method list called **default**. For most of the authentication and accounting types, the only method list that can be defined is default.

As soon as the default method list is defined for a given authentication or accounting type, it is automatically applied as the method list to be used for any instance of that type of authentication or accounting, except for instances to which another named method list has been specifically applied.

Configuring AAA Login Authentication

To configure AAA authentication, create default or a named method list for different authentication types. In the case of login authentication, the named method lists are then applied to consoles or VTY lines.

AAA Configuration Tasks

To define how a given accounting or authentication type will be applied to a given port or line:

- either create a server group using the **aaa group server** command (RADIUS only),
- or create a method list for the authentication or accounting type as required,
- then apply that method list to the port or line as required.

Step 1: Define a group of RADIUS servers

Create a server group using the **aaa group server** command.

To create a RADIUS server group named `GROUP1` with hosts `192.168.1.1`, `192.168.2.1` and `192.168.3.1`, use the commands:

```
awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# server 192.168.1.1 auth-port 1812 acct-port 1813
awplus(config-sg)# server 192.168.2.1 auth-port 1812 acct-port 1813
awplus(config-sg)# server 192.168.3.1 auth-port 1812 acct-port 1813
```

Step 2: Specify the login authentication or accounting method list

Create a method list for the authentication (**aaa authentication login**) or accounting (**aaa accounting login**) type as required.

To configure a user login authentication method list called `USERS` to use first all available RADIUS servers for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login default group radius local
```

To configure RADIUS accounting for login shell sessions, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa accounting login default start-stop group radius
```

To configure a user login authentication method list called USERS to use first the TACACS+ servers for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login USERS group tacacs+
local
```

Step 3: Apply method list to interface port or line

Apply that method list to the port or line as required.

```
awplus# configure terminal
awplus(config)# line console 0
awplus(config-line)# login authentication USERS
```

For most Authentication and Accounting types, the only possible server list is **default**, and the only server that can be put into it is **radius**. You will typically use all RADIUS servers, so **group radius** can be used, rather than having to create a specific user group. Often the configuration of a given Authentication or Accounting type will consist of a single command, the command that defines the default server list, which contains just one server.

AAA 802.1X-Authentication configuration

AAA 802.1X-Authentication will typically be configured with the following commands.

To enable 802.1X-Authentication globally for all RADIUS servers, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication dot1x default group radius
```

Sample Authentication Configurations

Sample 802.1X-Authentication Configuration

See the below sample configuration script for a sample 802.1X-Authentication configuration. Copy and paste then edit the sample 802.1X-Authentication configuration in your config file. See the **edit** command in the **Chapter 7, File Management Commands** for further information.

Output

Figure 66-1: Sample 802.1X-Authentication configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication dot1x default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user guest password guest!
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 dot1x port-control auto
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The 802.1X-Authentication feature needs the **aaa authentication dot1x** command and the **dot1x port-control** command configured on an interface. See **Chapter 67, AAA Commands** and **Chapter 69, 802.1X Commands** for command information to edit this configuration.

The local RADIUS Server has been configured to use 802.1X-Authentication in this sample configuration. See the **radius-server local** and **server enable** commands in **Chapter 73, Local RADIUS Server Commands** for command information to edit this sample configuration.

This sample configuration enables 802.1X-Authentication on interface `vlan1` with IP address `192.168.1.120`. Change the VLAN ID and IP address as required for your configuration.

Sample MAC-Authentication Configuration

See the below sample configuration script for a sample MAC-Authentication configuration. Copy, paste, and edit the sample MAC-Authentication configuration in the config file.

See the **edit** command in the **Chapter 7, File Management Commands** for further information.

Output

Figure 66-2: Sample MAC-Authentication configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication auth-mac default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user 00-d0-59-ab-70-37 password 00-d0-59-ab-70-37
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 auth-mac enable
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The MAC-Authentication feature needs the **aaa authentication auth-mac** and the **auth-mac enable** commands configured on an interface. See **Chapter 67, AAA Commands** and **Chapter 65, Authentication Commands** for command information to edit this configuration.

The local RADIUS Server has been configured to use MAC-Authentication in this sample configuration. See the **radius-server local** and **server enable** commands in **Chapter 73, Local RADIUS Server Commands** for command information to edit this sample configuration.

See the **user (RADIUS server)** command in **Chapter 73, Local RADIUS Server Commands** for command information to edit the MAC address of the supplicant for use with local RADIUS server as the RADIUS user name and the user password, as shown in the above configuration.

This configuration enables MAC-Authentication on `vlan1` with IP address `192.168.1.120`. Change the interface VLAN ID, MAC, and IP addresses as needed in your configuration.

Sample Web-Authentication Configuration

See the below sample configuration script for a sample Web-Authentication configuration. Copy, paste, and edit the sample Web-Authentication configuration for your config file.

See the **edit** command in the **Chapter 7, File Management Commands** for further information.

Output

Figure 66-3: Sample Web-Authentication configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication auth-web default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user guest encrypted password
 l+lWcLjLm29bCAXwWRPHXK0PFlsA7gNpR+P7wO4kwQQ=
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 auth-web enable
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The Web-Authentication feature needs the **aaa authentication auth-web** and the **auth-web enable** commands configured on an interface. See **Chapter 67, AAA Commands** and **Chapter 65, Authentication Commands** for command information to edit this configuration.

The local RADIUS Server has been configured to use Web-Authentication in this sample configuration. See the **radius-server local** and **server enable** commands in **Chapter 73, Local RADIUS Server Commands** for command information to edit this sample configuration.

The above sample Web-Authentication configuration requires the user name '`guest`' with password '`guest!`' on IP address `192.168.1.120` from interface `port1.0.1`.

Sample Tri-Authentication Configuration

See the below sample configuration script for a sample tri-authentication configuration that configures 802.1X-Authentication, MAC-Authentication, and Web-Authentication on the same interface. Copy, paste, and edit the sample tri-authentication configuration for your config file. See the **edit** command in the **Chapter 7, File Management Commands** for further information.

Output

Figure 66-4: Sample tri-authentication configuration

```
!
 radius-server host 127.0.0.1 key awplus-local-radius-server
!
 aaa authentication dot1x default group radius
 aaa authentication auth-mac default group radius
 aaa authentication auth-web default group radius
!
 radius-server local
 server enable
 nas 127.0.0.1 key awplus-local-radius-server
 user guest password guest!
 user 00-d0-59-ab-70-37 password 00-d0-59-ab-70-37
!
 no spanning-tree rstp enable
!
 interface port1.0.1
 switchport
 switchport mode access
 dot1x port-control auto
 auth-mac enable
 auth-web enable
!
 interface vlan1
 ip address 192.168.1.120/24
!
```

The 802.1X-Authentication feature needs the **aaa authentication dot1x** command and the **dot1x port-control** command configured on an interface. See **Chapter 67, AAA Commands** and **Chapter 69, 802.1X Commands** for command information to edit this configuration.

The MAC-Authentication feature needs the **aaa authentication auth-mac** and the **auth-mac enable** commands configured on an interface. See **Chapter 67, AAA Commands** and **Chapter 65, Authentication Commands** for command information to edit this configuration.

The Web-Authentication feature needs the **aaa authentication auth-web** and the **auth-web enable** commands configured on an interface. See **Chapter 67, AAA Commands** and **Chapter 65, Authentication Commands** for command information to edit this configuration.

The local RADIUS Server has been configured to use tri-authentication in this sample configuration. See the **radius-server local** and **server enable** commands in **Chapter 73, Local RADIUS Server Commands** for command information to edit this sample configuration.

This sample tri-authentication configuration requires a user name '`guest`' with password '`guest!`' on IP address `192.168.1.120` from `port1.0.1`. Note this sample also configures 802.1X- and MAC-Authentication on `vlan1` with IP address `192.168.1.120`. Change the interface VLAN ID, MAC and IP address as needed for your configuration.

Note that when tri-authentication is applied to the same interface then the order of execution is MAC-Authentication first, then 802.1X- or Web-Authentication, if MAC-Authentication fails.

Chapter 67: AAA Commands

Command List 67.2
aaa accounting auth-mac default 67.2
aaa accounting auth-web default 67.4
aaa accounting commands 67.6
aaa accounting dot1x 67.8
aaa accounting login 67.10
aaa accounting update 67.12
aaa authentication auth-mac 67.13
aaa authentication auth-web 67.14
aaa authentication dot1x 67.15
aaa authentication enable default group tacacs+ 67.16
aaa authentication enable default local 67.18
aaa authentication login 67.19
aaa group server 67.22
aaa local authentication attempts lockout-time 67.23
aaa local authentication attempts max-fail 67.24
accounting login 67.25
clear aaa local user lockout 67.26
debug aaa 67.27
login authentication 67.28
show debugging aaa 67.29
undebug aaa 67.29

Command List

This chapter provides an alphabetical reference for AAA commands for Authentication, Authorization and Accounting. For more information, see **Chapter 66, AAA Introduction and Configuration**.

aaa accounting auth-mac default

This command configures a default accounting method list for MAC-based Authentication. The default accounting method list specifies what type of accounting messages are sent and specifies which RADIUS Servers the accounting messages are sent to. The default accounting method list is automatically applied to interfaces with MAC-based Authentication enabled.

Use the **no** variant of this command to disable AAA accounting for MAC-based Authentication globally.

Syntax

```
aaa accounting auth-mac default {start-stop|stop-only|none}
   group {<group-name>|radius}

no aaa accounting auth-mac default
```

Parameter	Description
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
`<group-name>`	Server group name.
`radius`	Use all RADIUS servers

Default RADIUS accounting for MAC-based Authentication is disabled by default

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command
- **group <*group-name*>** : use the specified RADIUS server group configured with the **aaa group server** command

The accounting event to send to the RADIUS server is configured with the following options:

- **start-stop** : sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only** : sends a **stop** accounting message at the end of a session.
- **none** : disables accounting.

Examples To enable RADIUS accounting for MAC-based Authentication, and use all available RADIUS Servers, use the commands:

```
awplus# configure terminal
awplus(config)# aaa accounting auth-mac default start-stop
group radius
```

To disable RADIUS accounting for MAC-based Authentication, use the commands:

```
awplus# configure terminal
awplus(config)# no aaa accounting auth-mac default
```

Related Commands **aaa authentication auth-mac**

aaa accounting auth-web default

This command configures a default accounting method list for Web-based Port Authentication. The default accounting method list specifies what type of accounting messages are sent and specifies which RADIUS Servers the accounting messages are sent to. The default accounting method list is automatically applied to interfaces with Web-based Authentication enabled.

Use the **no** variant of this command to disable AAA accounting for Web-based Port Authentication globally.

Syntax

```
aaa accounting auth-web default {start-stop|stop-only|none}
   group {<group-name>|radius}

no aaa accounting auth-web default
```

Parameter	Description
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
<group-name>	Server group name.
`radius`	Use all RADIUS servers.

Default RADIUS accounting for Web-based Port Authentication is disabled by default.

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command
- **group <*group-name*>** : use the specified RADIUS server group configured with the **aaa group server** command

Configure the accounting event to be sent to the RADIUS server with the following options:

- **start-stop** : sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only** : sends a **stop** accounting message at the end of a session.
- **none** : disables accounting.

Examples To enable RADIUS accounting for Web-based Authentication, and use all available RADIUS Servers, use the commands:

```
awplus# configure terminal
awplus(config)# aaa accounting auth-web default start-stop
group radius
```

To disable RADIUS accounting for Web-based Authentication, use the commands:

```
awplus# configure terminal
awplus(config)# no aaa accounting auth-web default
```

Related Commands **aaa authentication auth-web**

aaa accounting commands

Use this command to configure and enable TACACS+ command accounting. When command accounting is enabled, information about a command entered at a specified privilege level on a device is sent to a TACACS+ server. To account for all commands entered on a device you need to configure command accounting for each discrete privilege level. A command accounting record includes the command as entered for the specified privilege level, the date and time each command execution finished, and the username of the user who executed the command.

This command creates a default method list that is applied to every console and vty line. The **stop-only** parameter indicates that an accounting message is sent to the TACACS+ server when a command has stopped executing.

Note that up to four TACACS+ servers can be configured for accounting. The servers are checked for reachability in the order they are configured and only the first reachable server is used. If no server is found the accounting message is dropped.

Use the **no** variant of this command to disable command accounting.

Syntax

```
aaa accounting commands <1-15> default stop-only group tacacs+
no aaa accounting commands <1-15> default
```

Parameter	Description
<1-15>	The privilege level, in the range 1 to 15.

Default TACACS+ command accounting is disabled by default.

Mode Global Configuration

Usage When command accounting is enabled, the command as entered is included in the accounting packets sent to the TACACS+ accounting server.

You cannot enable command accounting if a trigger is configured. An error message is displayed if you attempt to enable command accounting and a trigger is configured.

The **show tech-support** command runs a number of commands and each command is accounted separately.

When the **copy <*filename*> running-config** command is executed all the commands of a configuration file copied into the running-config are accounted separately.

Examples To configure command accounting for privilege level 15 commands, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa accounting commands 15 default stop-only
group tacacs+
```

To disable command accounting for privilege level 15 commands, use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa accounting commands 15 default
```

Related Commands

aaa authentication login
aaa accounting login
accounting login
tacacs-server host

aaa accounting dot1x

This command configures the default accounting method list for IEEE 802.1X-based Authentication. The default accounting method list specifies what type of accounting messages are sent and specifies which RADIUS Servers the accounting messages are sent to. The default accounting method list is automatically applied to interfaces with IEEE 802.1X-based Authentication enabled.

Use the **no** variant of this command to disable AAA accounting for 802.1X-based Port Authentication globally.

Syntax

```
aaa accounting dot1x default {start-stop|stop-only|none}
   group {<group-name>|radius}

no aaa accounting dot1x default
```

Parameter	Description
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
<group-name>	Server group name.
`radius`	Use all RADIUS servers.

Default RADIUS accounting for 802.1X-based Port Authentication is disabled by default (there is no default server set by default).

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages will be sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command.
- **group <*group-name*>** : use the specified RADIUS server group configured with the **aaa group server** command.

The accounting event to send to the RADIUS server is configured by the following options:

- **start-stop** : sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only** : sends a **stop** accounting message at the end of a session.
- **none** : disables accounting.

Examples To enable RADIUS accounting for 802.1X-based Authentication, and use all available RADIUS Servers, use the commands:

```
awplus# configure terminal
awplus(config)# aaa accounting dot1x default start-stop group
radius
```

To disable RADIUS accounting for 802.1X-based Authentication, use the commands:

```
awplus# configure terminal
awplus(config)# no aaa accounting dot1x default
```

Related Commands

aaa accounting update
aaa authentication dot1x
aaa group server
dot1x port-control
radius-server host

aaa accounting login

This command configures RADIUS and TACACS+ accounting for login shell sessions. The specified method list name can be used by the **accounting login** command in the Line Configuration mode. If the **default** parameter is specified, then this creates a default method list that is applied to every console and vty line, unless another accounting method list is applied on that line.

Note that unlimited RADIUS servers and up to four TACACS+ servers can be configured and consulted for accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn. A backup server is consulted if the primary server fails, i.e. is unreachable.

Use the **no** variant of this command to remove an accounting method list for login shell sessions configured by an **aaa accounting login** command. If the method list being deleted is already applied to a console or vty line, accounting on that line will be disabled. If the default method list name is removed by this command, it will disable accounting on every line that has the default accounting configuration.

Syntax

```
aaa accounting login {default|<list-name>}{start-stop|stop-only|none}
    {group {radius|tacacs+|<group-name>}}

no aaa accounting login {default|<list-name>}
```

Parameter	Description
`default`	Default accounting method list.
`<list-name>`	Named accounting method list.
`start-stop`	Start and stop records to be sent.
`stop-only`	Stop records to be sent.
`none`	No accounting record to be sent.
`group`	Specify the servers or server group where accounting packets are sent.
`radius`	Use all RADIUS servers configured by the **radius-server host** command on page 71.6.
`tacacs+`	Use all TACACS+ servers configured by the **tacacs-server host** command.
`<group-name>`	Use the specified RADIUS server group, as configured by the **aaa group server** command.

Default Accounting for login shell sessions is disabled by default.

Mode Global Configuration

Usage This command enables you to define a named accounting method list. The items that you define in the accounting options are:

- the types of accounting packets that will be sent
- the set of servers to which the accounting packets will be sent

You can define a default method list with the name `default` and any number of other named method lists. The *<list-name>* for any method list that you define can then be used as the *<list-name>* parameter in the **accounting login** command available from Line Configuration mode.

If the method list name already exists, the command will replace the existing configuration with the new one.

There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command
- **group <*group-name*>** : use the specified RADIUS server group configured with the **aaa group server** command

There is one way to define servers where TACACS+ accounting messages are sent:

- **group tacacs+** : use all TACACS+ servers configured by **tacacs-server host** command

The accounting event to send to the RADIUS or TACACS+ server is configured with the following options:

- **start-stop** : sends a **start** accounting message at the beginning of a session and a **stop** accounting message at the end of the session.
- **stop-only** : sends a **stop** accounting message at the end of a session.
- **none** : disables accounting.

Examples To configure RADIUS accounting for login shell sessions, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa accounting login default start-stop group
radius
```

To configure TACACS+ accounting for login shell sessions, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa accounting login default start-stop group
tacacs+
```

To reset the configuration of the default accounting list, use the following commands:

```
awplus# configure terminal
awplus(config)#  no aaa accounting login default
```

Related Commands
aaa accounting commands
aaa authentication login
aaa accounting login
aaa accounting update
accounting login
radius-server host
tacacs-server host

aaa accounting update

This command enables periodic accounting reporting to either the RADIUS or TACACS+ accounting server(s) wherever login accounting has been configured.

Note that unlimited RADIUS servers and up to four TACACS+ servers can be configured and consulted for accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn. A backup server is consulted if the primary server fails, i.e. is unreachable.

Use the **no** variant of this command to disable periodic accounting reporting to the accounting server(s).

Syntax

```
aaa accounting update [periodic <1-65535>]
no aaa accounting update
```

Parameter	Description
`periodic`	Send accounting records periodically.
<1-65535>	The interval to send accounting updates (in minutes). The default is 30 minutes.

Default Periodic accounting update is disabled by default.

Mode Global Configuration

Usage Use this command to enable the device to send periodic AAA login accounting reports to the accounting server. When periodic accounting report is enabled, interim accounting records are sent according to the interval specified by the **periodic** parameter. The accounting updates are start messages.

If the **no** variant of this command is used to disable periodic accounting reporting, any interval specified by the **periodic** parameter is reset to the default of 30 minutes when accounting reporting is reenabled, unless this interval is specified.

Examples To configure the switch to send period accounting updates every 30 minutes, the default period, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa accounting update
```

To configure the switch to send period accounting updates every 10 minutes, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa accounting update periodic 10
```

To disable periodic accounting update wherever accounting has been configured, use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa accounting update
```

Related Commands
aaa accounting auth-mac default
aaa accounting auth-web default
aaa accounting dot1x
aaa accounting login

aaa authentication auth-mac

This command enables MAC-based Port Authentication globally and allows you to specify an authentication method list. It is automatically applied to every interface running MAC-based Port Authentication.

Use the **no** variant of this command to globally disable MAC-based Port Authentication.

Syntax

```
aaa authentication auth-mac default group {<group-name>|radius}
no aaa authentication auth-mac default
```

Parameter	Description
<group-name>	Server group name.
radius	Use all RADIUS servers.

Default MAC-based Port Authentication is disabled by default.

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command
- **group <*group-name*>** : use the specified RADIUS server group configured with the **aaa group server** command

All configured RADIUS Servers are automatically members of the server group **radius**. If a server is added to a named group **<*group-name*>**, it also remains a member of the group **radius**.

Examples To enable MAC-based Port Authentication globally for all RADIUS servers, and use all available RADIUS servers, use the commands:

```
awplus# configure terminal
awplus(config)# aaa authentication auth-mac default group
radius
```

To disable MAC-based Port Authentication, use the commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication auth-mac default
```

Related Commands
aaa accounting auth-mac default
auth-mac enable

aaa authentication auth-web

This command enables Web-based Port Authentication globally and allows you to enable an authentication method list (in this case, a list of RADIUS Servers). It is automatically applied to every interface running Web-based Port Authentication.

Use the **no** variant of this command to globally disable Web-based Port Authentication.

Syntax

```
aaa authentication auth-web default group {<group-name>|radius}
no aaa authentication auth-web default
```

Parameter	Description
<group-name>	Server group name.
radius	Use all RADIUS servers.

Default Web-based Port Authentication is disabled by default.

Mode Global Configuration

Usage There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command
- **group *<group-name>*** : use the specified RADIUS server group configured with the **aaa group server** command

Note that you need to configure an IPv4 address for the VLAN interface on which We Authentication is running.

Examples To enable Web-based Port Authentication globally for all RADIUS servers, and use all available RADIUS servers, use the commands:

```
awplus# configure terminal
awplus(config)# aaa authentication auth-web default group
radius
```

To disable Web-based Port Authentication, use the commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication auth-web default
```

Related Commands **aaa accounting auth-web default**
auth-mac enable

aaa authentication dot1x

This command enables 802.1X-based Port Authentication globally and allows you to enable an authentication method list. It is automatically applied to every interface running 802.1X-based Port Authentication.

Use the **no** variant of this command to globally disable 802.1X-based Port Authentication.

Syntax

```
aaa authentication dot1x default group {<group-name>|radius}
no aaa authentication dot1x default
```

Parameter	Description
`radius`	Use all RADIUS servers.
<group-name>	Server group name.

Default 802.1X-based Port Authentication is disabled by default.

Mode Global Configuration

Usage Use this command to specify the default method list to use for authentication on all switch ports with 802.1X enabled. Use the **no** variant of this command to reset the authentication method list for 802.1X to its default (i.e. to use the group **radius**, containing all RADIUS servers configured by the **radius-server host** command).

There are two ways to define servers where RADIUS accounting messages are sent:

- **group radius** : use all RADIUS servers configured by **radius-server host** command
- **group *<group-name>*** : use the specified RADIUS server group configured with the **aaa group server** command

Examples To enable 802.1X-based Port Authentication globally with all RADIUS servers, and use all available RADIUS servers, use the command:

```
awplus# configure terminal
awplus(config)# aaa authentication dot1x default group radius
```

To disable 802.1X-based Port Authentication, use the command:

```
awplus# configure terminal
awplus(config)# no aaa authentication dot1x default
```

Related Commands

aaa accounting dot1x
aaa group server
dot1x port-control
radius-server host

aaa authentication enable default group tacacs+

This command enables AAA authentication to determine the privilege level a user can access for passwords authenticated against the TACACS+ server.

Use the **no** variant of this command to disable privilege level authentication.

Syntax

```
aaa authentication enable default group tacacs+ [local] [none]
no aaa authentication enable default
```

Parameter	Description
`local`	Use the locally configured enable password (**enable password** command) for authentication.
`none`	No authentication.

Default Local privilege level authentication is enabled by default (**aaa authentication enable default local** command).

Mode Global Configuration

Usage A user is configured on a TACACS+ server with a maximum privilege level. When they enter the **enable (Privileged Exec mode)** command they are prompted for an enable password which is authenticated against the TACACS+ server. If the password is correct and the specified privilege level is equal to or less than the users maximum privilege level, then they are granted access to that level. If the user attempts to access a privilege level that is higher than their maximum configured privilege level, then the authentication session will fail and they will remain at their current privilege level.

Note If both **local** and **none** are specified, you must always specify **local** first.

If the TACACS+ server goes offline, or is not reachable during enable password authentication, and command level authentication is configured as:

- **aaa authentication enable default group tacacs+**
  then the user is never granted access to Privileged Exec mode.
- **aaa authentication enable default group tacacs+ local**
  then the user is authenticated using the locally configured enable password, which if entered correctly grants the user access to Privileged Exec mode. If no enable password is locally configured (**enable password** command), then the enable authentication will fail until the TACACS+ server becomes available again.
- **aaa authentication enable default group tacacs+ none**
  then the user is granted access to Privileged Exec mode with no authentication. This is true even if a locally configured enable password is configured.
- **aaa authentication enable default group tacacs+ local none**
  then the user is authenticated using the locally configured enable password. If no enable password is locally configured, then the enable authentication will grant access to Privileged Exec mode with no authentication.

If the password for the user is not successfully authenticated by the server, then the user is again prompted for an enable password when they enter **enable** via the CLI.

Examples

To enable a privilege level authentication method that will not allow the user to access Privileged Exec mode if the TACACS+ server goes offline, or is not reachable during enable password authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication enable default group tacacs+
```

To enable a privilege level authentication method that will allow the user to access Privileged Exec mode if the TACACS+ server goes offline, or is not reachable during enable password authentication, and a locally configured enable password is configured, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication enable default group tacacs+
local
```

To disable privilege level authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication enable default
```

Related Commands

aaa authentication login
aaa authentication enable default local
enable (Privileged Exec mode)
enable password
enable secret
tacacs-server host

aaa authentication enable default local

This command enables AAA authentication to determine the privilege level a user can access for passwords authenticated locally.

Syntax `aaa authentication enable default local`

Default Local privilege level authentication is enabled by default.

Mode Global Configuration

Usage The privilege level configured for a particular user in the local user database is the privilege threshold above which the user is prompted for an **enable (Privileged Exec mode)** command.

Examples To enable local privilege level authentication command, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication enable default local
```

To disable privilege level authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication enable default
```

Related Commands
aaa authentication enable default group tacacs+
aaa authentication login
enable (Privileged Exec mode)
enable password
enable secret
tacacs-server host

aaa authentication login

Use this command to create an ordered list of methods to use to authenticate user login, or to replace an existing method list with the same name. Specify one or more of the options **local** or **group**, in the order you want them to be applied. If the **default** method list name is specified, it is applied to every console and VTY line immediately unless another method list is applied to that line by the **login authentication** command. To apply a non-default method list, you must also use the **login authentication** command.

Use the **no** variant of this command to remove an authentication method list for user login. The specified method list name is deleted from the configuration. If the method list name has been applied to any console or VTY line, user login authentication on that line will fail.

Note that the **no aaa authentication login default** command does not remove the default method list. This will return the default method list to its default state (**local** is the default).

Syntax

```
aaa authentication login {default|<list-name>}
   {[local] [group {radius|tacacs+|<group-name>}]}

no aaa authentication login {default|<list-name>}
```

Parameter	Description
`default`	Set the default authentication server for user login.
`<list-name>`	Name of authentication server.
`local`	Use the local username database.
`group`	Use server group.
`radius`	Use all RADIUS servers configured by the **radius-server host** command on page 71.6.
`tacacs+`	Use all TACACS+ servers configured by the **tacacs-server host** command.
`<group-name>`	Use the specified RADIUS server group, as configured by the **aaa group server** command.

Default If the default server is not configured using this command, user login authentication uses the local user database only.

If the **default** method list name is specified, it is applied to every console and VTY line immediately unless a named method list server is applied to that line by the **login authentication** command.

local is the default state for the default method list unless a named method list is applied to that line by the **login authentication** command. Reset to the default method list using the **no aaa authentication login default** command.

Mode Global Configuration

Usage When a user attempts to log in, the switch sends an authentication request to the first authentication server in the method list. If the first server in the list is reachable and it contains a username and password matching the authentication request, the user is authenticated and the login succeeds. If the authentication server denies the authentication request because of an incorrect username or password, the user login fails. If the first server in the method list is unreachable, the switch sends the request to the next server in the list, and so on.

For example, if the method list specifies group tacacs+ local, and a user attempts to log in with a password that does not match a user entry in the first TACACS+ server, if this TACACS+ server denies the authentication request, then the switch does not try any other TACACS+ servers not the local user database; the user login fails.

Examples To configure the default authentication method list for user login to use first all available RADIUS servers for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login default group radius
local
```

To configure a user login authentication method list called `USERS` to use first the RADIUS server group `RAD_GROUP1` for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login USERS group
RAD_GROUP1 local
```

To configure a user login authentication method list called `USERS` to use first the TACACS+ servers for user login authentication and then the local user database, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa authentication login USERS group tacacs+
local
```

To return to the default method list (**local** is the default server), use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication login default
```

To delete an existing authentication method list `USERS` created for user login authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# no aaa authentication login USERS
```

Related Commands

aaa accounting commands
aaa authentication enable default group tacacs+
login authentication

aaa group server

This command configures a RADIUS server group. A server group can be used to specify a subset of RADIUS servers in **aaa** commands. The group name **radius** is predefined, which includes all RADIUS servers configured by the **radius-server host** command.

RADIUS servers are added to a server group using the **server** command. Each RADIUS server should be configured using the **radius-server host** command.

Use the **no** variant of this command to remove an existing RADIUS server group.

Syntax

```
aaa group server radius <group-name>
no aaa group server radius <group-name>
```

Parameter	Description
<group-name>	Server group name.

Mode Global Configuration

Usage Use this command to create an AAA group of RADIUS servers, and to enter Server Group Configuration mode, in which you can add servers to the group. Use a server group to specify a subset of RADIUS servers in AAA commands. Each RADIUS server must be configured by the **radius-server host** command. To add RADIUS servers to a server group, use the **server** command.

Examples To create a RADIUS server group named `GROUP1` with hosts `192.168.1.1`, `192.168.2.1` and `192.168.3.1`, use the commands:

```
awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# server 192.168.1.1 auth-port 1812 acct-port 1813
awplus(config-sg)# server 192.168.2.1 auth-port 1812 acct-port 1813
awplus(config-sg)# server 192.168.3.1 auth-port 1812 acct-port 1813
```

To remove a RADIUS server group named `GROUP1` from the configuration, use the command:

```
awplus(config)# no aaa group server radius GROUP1
```

Related Commands

aaa accounting auth-mac default
aaa accounting auth-web default
aaa accounting dot1x
aaa accounting login
aaa authentication auth-mac
aaa authentication auth-web
aaa authentication dot1x
aaa authentication login
radius-server host
server (Server Group)

aaa local authentication attempts lockout-time

This command configures the duration of the user lockout period.

Use the **no** variant of this command to restore the duration of the user lockout period to its default of 300 seconds (5 minutes).

Syntax

```
aaa local authentication attempts lockout-time <lockout-tiime>
no aaa local authentication attempts lockout-time
```

Parameter	Description
<lockout-time>	<0-10000>. Time in seconds to lockout the user.

Mode Global Configuration

Default The default for the lockout-time is 300 seconds (5 minutes).

Usage While locked out all attempts to login with the locked account will fail. The lockout can be manually cleared by another privileged account using the **clear aaa local user lockout** command.

Examples To configure the lockout period to 10 minutes (600 seconds), use the commands:

```
awplus# configure terminal
awplus(config)# aaa local authentication attempts lockout-time
600
```

To restore the default lockout period of 5 minutes (300 seconds), use the commands:

```
awplus# configure terminal
awplus(config)# no aaa local authentication attempts
lockout-time
```

Related Commands **aaa local authentication attempts max-fail**

aaa local authentication attempts max-fail

This command configures the maximum number of failed login attempts before a user account is locked out. Every time a login attempt fails the failed login counter is incremented.

Use the **no** variant of this command to restore the maximum number of failed login attempts to the default setting (five failed login attempts).

Syntax

```
aaa local authentication attempts max-fail <failed-logins>
no aaa local authentication attempts max-fail
```

Parameter	Description
<failed-logins>	<1-32>. Number of login failures allowed before locking out a user.

Mode Global Configuration

Default The default for the maximum number of failed login attempts is five failed login attempts.

Usage When the failed login counter reaches the limit configured by this command that user account is locked out for a specified duration configured by the **aaa local authentication attempts lockout-time** command.

When a successful login occurs the failed login counter is reset to 0. When a user account is locked out all attempts to login using that user account will fail.

Examples To configure the number of login failures that will lock out a user account to two login attempts, use the commands:

```
awplus# configure terminal
awplus(config)# aaa local authentication attempts max-fail 2
```

To restore the number of login failures that will lock out a user account to the default number of login attempts (five login attempts), use the commands:

```
awplus# configure terminal
awplus(config)# no aaa local authentication attempts max-fail
```

Related Commands **aaa local authentication attempts lockout-time**
clear aaa local user lockout

accounting login

This command applies a login accounting method list to console or vty lines for user login. When login accounting is enabled using this command, logging events generate an accounting record to the accounting server.

The accounting method list must be configured first using this command. If an accounting method list is specified that has not been created by this command then accounting will be disabled on the specified lines.

The **no** variant of this command resets AAA (Authentication, Authorization, Accounting) Accounting applied to console or vty lines for local or remote login. **default** login accounting is applied after issuing the **no accounting login** command. Accounting is disabled with **default**.

Syntax

```
accounting login {default|<list-name>}
no accounting login
```

Parameter	Description
`default`	Default accounting method list.
<list-name>	Named accounting method list.

Default By default login accounting is disabled in the **default** accounting server. No accounting will be performed until accounting is enabled using this command beforehand.

Mode Line Configuration

Examples To apply the accounting server `USERS` to all vty lines use the following commands:

```
awplus# configure terminal
awplus(config)# line vty 0 32
awplus(config-line)# accounting login USERS
```

Related Commands
aaa accounting commands
aaa accounting login

clear aaa local user lockout

Use this command to clear the lockout on a specific user account or all user accounts.

Syntax `clear aaa local user lockout {username <username>|all}`

Parameter	Description
`username`	Clear lockout for the specified user.
`<username>`	Specifies the user account.
`all`	Clear lockout for all user accounts.

Mode Privileged Exec

Examples To unlock the user account '`bob`' use the following command:

```
awplus# clear aaa local user lockout username bob
```

To unlock all user accounts use the following command:

```
awplus# clear aaa local user lockout all
```

Related Commands **aaa local authentication attempts lockout-time**

debug aaa

This command enables AAA debugging.

Use the **no** variant of this command to disable AAA debugging.

Syntax

```
debug aaa [accounting|all|authentication|authorization]
no debug aaa [accounting|all|authentication|authorization]
```

Parameter	Description
`accounting`	Accounting debugging.
`all`	All debugging options are enabled.
`authentication`	Authentication debugging.
`authorization`	Authorization debugging.

Default AAA debugging is disabled by default.

Mode Privileged Exec

Examples To enable authentication debugging for AAA, use the command:

```
awplus# debug aaa authentication
```

To disable authentication debugging for AAA, use the command:

```
awplus# no debug aaa authentication
```

Related Commands **show debugging aaa**
undebug aaa

login authentication

Use this command to apply an AAA server for authenticating user login attempts from a console or remote logins on these console or VTY lines. The authentication method list must be specified by the **aaa authentication login** command. If the method list has not been configured by the **aaa authentication login** command, login authentication will fail on these lines.

Use the **no** variant of this command to reset AAA Authentication configuration to use the default method list for login authentication on these console or VTY lines.

Command Syntax

```
login authentication {default|<list-name>}
no login authentication
```

Parameter	Description
`default`	The default authentication method list. If the default method list has not been configured by the **aaa authentication login** command, the local user database is used for user login authentication.
<list-name>	Named authentication server.

Default The default login authentication method list, as specified by the **aaa authentication login** command, is used to authenticate user login. If this has not been specified, the default is to use the local user database.

Mode Line Configuration

Examples To reset user authentication configuration on all VTY lines, use the following commands:

```
awplus# configure terminal
awplus(config)# line vty 0 32
awplus(config-line)# no login authentication
```

Related Commands

aaa authentication login
line

show debugging aaa

This command displays the current debugging status for AAA (Authentication, Authorization, Accounting).

Syntax `show debugging aaa`

Mode User Exec and Privileged Exec

Example To display the current debugging status of AAA, use the command:

```
awplus# show debug aaa
```

Output **Figure 67-1: Example output from the show debug aaa command**

```
AAA debugging status:
    Authentication debugging is on
    Accounting debugging is off
```

undebug aaa

This command applies the functionality of the **no debug aaa** command on page 67.27.

Chapter 68: 802.1X Introduction and Configuration

Introduction 68.2
802.1X System Components 68.2
The 802.1X Implementation 68.5
Configuring 802.1X 68.7

Introduction

802.1X is an IEEE standard providing a mechanism for authenticating devices attached to a LAN port or wireless device. Devices wishing to access services behind a port must authenticate themselves before any Ethernet packets are allowed to pass through. The protocol is referred to as 802.1X because it was initially defined in the IEEE standard 802.1X, published in 2001 and revised in 2004 and again as the current 802.1X 2010 standard.

Networks have two important requirements:

- Security: Authentication and Authorization
- Flexibility: The ability for users to roam

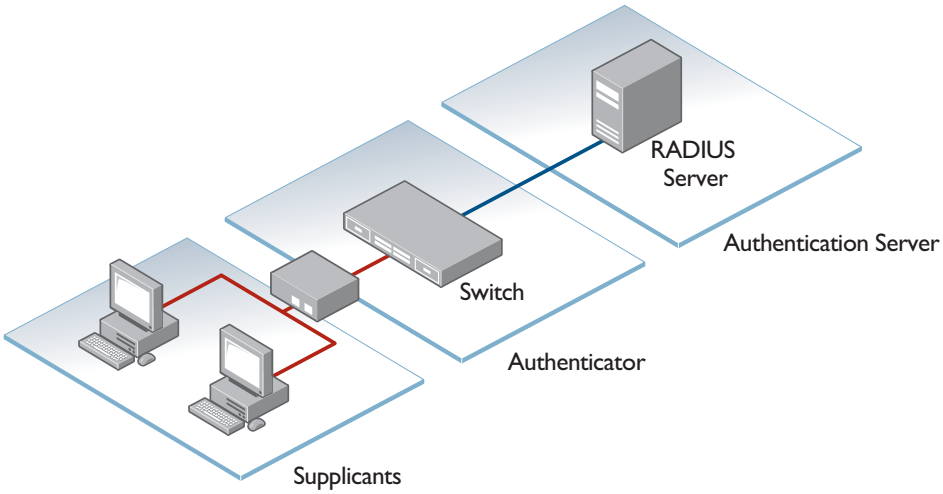
Networks need a device authentication method that is highly secure, but not tied to a port's physical location. Network resources presented to a given user need to be determined from their authentication credentials.

802.1X user authentication satisfies these requirements. It is relatively uncomplicated and has little impact on network performance. It is a protocol that is medium-independent — being equally as effective on wireless connections (802.11i) and wired connections. 802.1X user authentication is rapidly becoming an expected component on networks.

802.1X System Components

There are three main components to a system using 802.1X port authentication control:

- Authenticator: The device that wishes to enforce authentication before allowing access to services that are accessible behind it. An example of this is a switch that has 802.1X port authentication control enabled.
- Supplicant: The client that wishes to access services offered by the authenticator's system. An example of this is a Windows XP Professional PC with an 802.1X client.
- Authentication server: The device that uses the authentication credentials supplied by the supplicant, to determine if the authenticator should grant access to its services. The AlliedWare Plus implementation of 802.1X supports the use of a RADIUS. authentication server using Extensible Authentication Protocol (EAP) in conjunction with RADIUS.

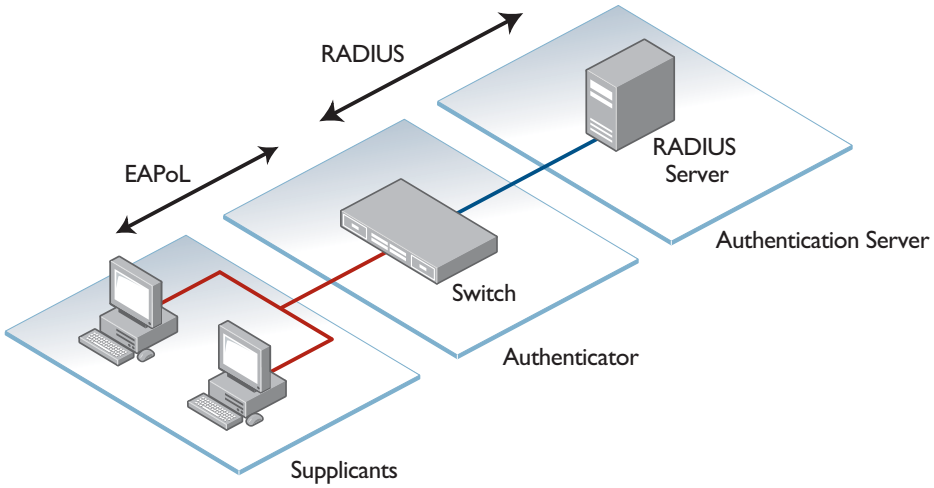


802.1X component protocols

There are two protocols involved in the authentication conversation:

- EAPoL exchanged between the supplicant and authenticator.
  - « EAPoL—Extensible Authentication Protocol over LAN— is the protocol defined in IEEE802.1X.
- RADIUS exchanged between the authenticator and authentication server.
  - « RADIUS has received specific extensions to interoperate with EAPoL.

The diagram below illustrates where EAPoL and RADIUS protocols are used in the authentication conversation:



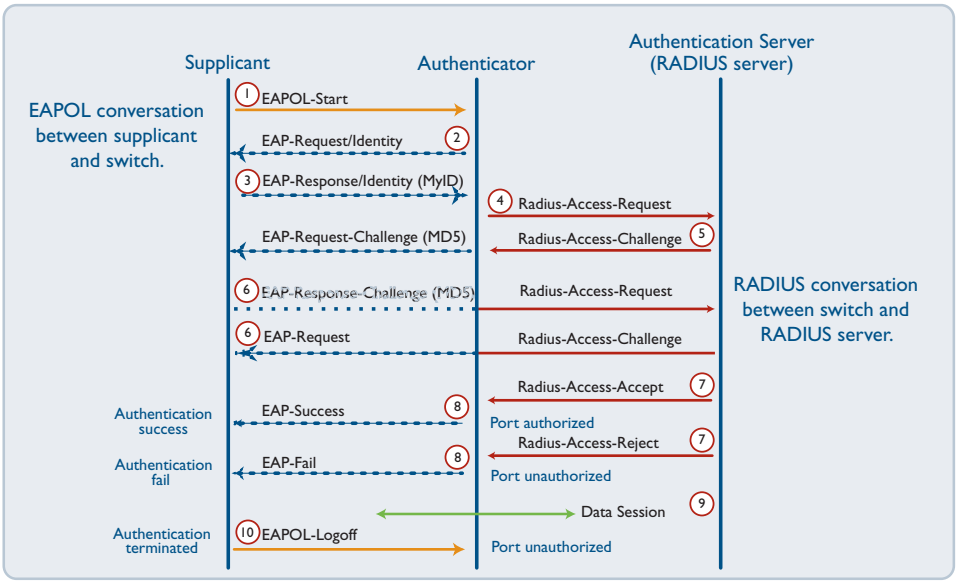
Basic steps in an 802.1X conversation

Step	Action
1	The supplicant informs the authenticator that it wants to initiate the conversation.
2	The authenticator requests the supplicant's credentials.
3	The supplicant sends username/password or X.509 certificate.
4	The authenticator wraps the supplicant's reply into a RADIUS packet and sends it to the RADIUS server.
5	The RADIUS server chooses an authentication method, and sends an appropriate request to the supplicant as a 'challenge'.
6	The RADIUS server and supplicant exchange some messages, ferried by the authenticator.
7	The RADIUS server eventually decides if the supplicant is allowed access and the RADIUS server sends an Access-Accept or Access-Reject message to the Authenticator.
8	The authenticator sends an EAPoL-Success or EAPoL-Fail to the supplicant.
9	The supplicant has a session using the network (if accepted).
10	When the session is over, the supplicant sends a log-off message.

Example message sequence

The diagram below illustrates an exchange using the EAP-MD5 authentication method, which is the simplest authentication method supported by 802.1X.

The EAPoL log-off message, of course, is not sent immediately after the other messages in the diagram, but is sent later on, at the end of the supplicant's data session, when it wishes to disconnect from the network.The EAPoL log-off message, of course, is not sent immediately after the other messages in the diagram, but is sent later on, at the end of the supplicant's data session, when it wishes to disconnect from the network.



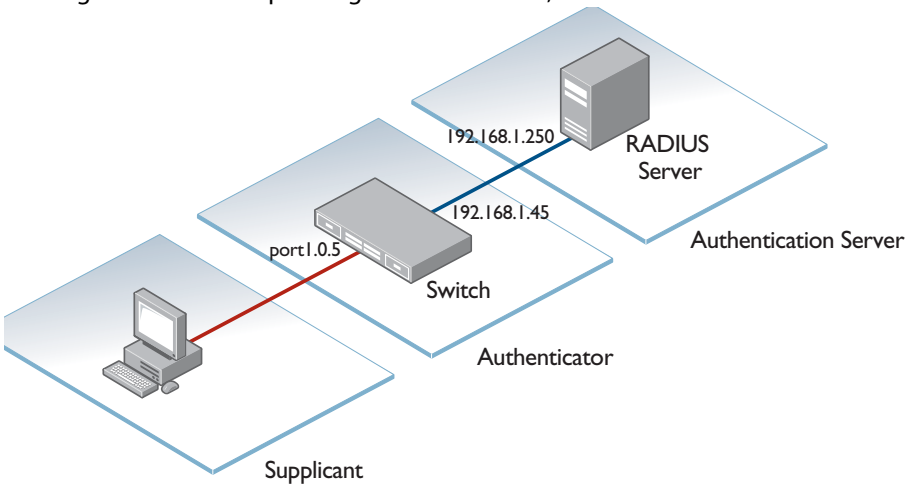
The 802.1X Implementation

802.1X port access control is achieved by making devices attached to a controlled port authenticate themselves via communication with an authentication server before these devices are allowed to access the network behind the controlled port.

Authentication is required on a per-port basis. The main components of an 802.1X implementation are:

- the authenticator - the port on this device that wishes to enforce authentication before allowing access to services that are accessible behind it.
- the supplicant - the port that wishes to access services offered by the authenticator's system. The supplicant may be a port on a PC or other device connected to this device.
- the authentication server - a device that uses the authentication credentials supplied by the supplicant, via the authenticator, to determine if the authenticator should grant access to its services.

To configure the switch operating as authenticator, follow the instructions below:



1. Configure a RADIUS server for the switch to send requests to

```
awplus(config)# radius-server host 192.168.1.250 key
                <secret-key>
```

2. Instruct 802.1X to use the configured RADIUS server

```
awplus(config)# aaa authentication dot1x default group radius
```

3. Configure port1.0.5 for 802.1X authentication

```
   awplus(config)# interface port1.0.5
awplus(config-if)# dot1x port-control auto
awplus(config-if)# spanning-tree portfast
```

Configuring 802.1X

The following example explains how to configure 802.1X. In this example, the RADIUS Server keeps the Client information, validating the identity of the Client and updating the switch about the authentication status of the client. The switch is the physical access between the two clients and the server. It requests information from the client, relays information to the server and then back to the client.

To configure 802.1X authentication, first enable authentication on `port1.0.1` and `port1.0.2` and then specify the RADIUS Server IP address and port.

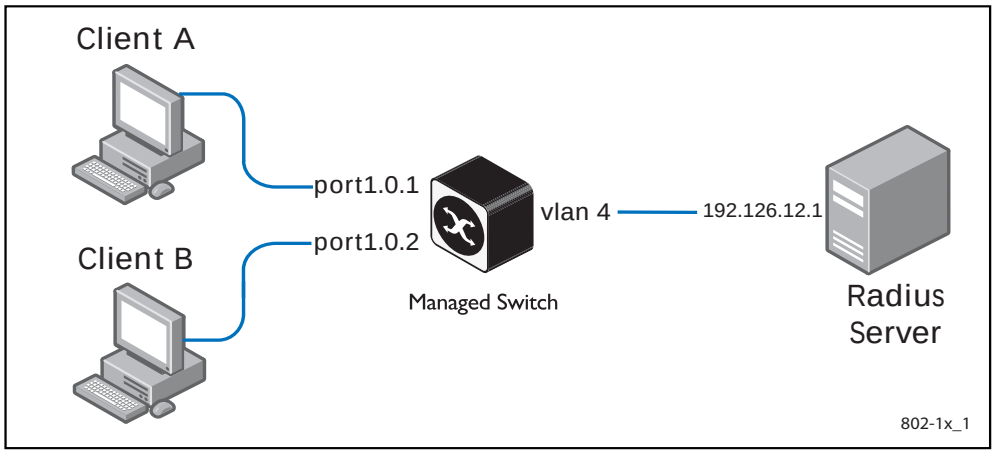


Table 68-1: 802.1X configuration on the switch

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `aaa authentication dot1x default group radius`	Enable authentication globally.
`awplus(config)#` `interface port1.0.1`	Specify the interface (`port1.0.1`) to be configured and enter the Interface mode.
`awplus(config-if)#` `dot1x port-control auto`	Enable authentication (via RADIUS) on `port1.0.1`.
`awplus(config-if)#` `dot1x control-direction both`	Block traffic in both directions, other than authentication packets, until authentication is complete.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.
`awplus(config)#` `interface port1.0.2`	Specify the interface (`port1.0.2`) you are configuring and enter the Interface mode.
`awplus(config-if)#` `dot1x port-control auto`	Enable authentication (via RADIUS) on `port1.0.2`.

Table 68-1: 802.1X configuration on the switch (cont.)

`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.
`awplus(config)#`	
`radius-server host 192.126.12.1 auth-port 1812`	Specify the RADIUS Server address (`192.126.12.1`) and authentication port.
`awplus(config)#`	
`radius-server key secret`	Specify the shared key `secret` between the RADIUS server and the client.
`awplus(config)#`	
`interface vlan4`	Specify the vlan (`vlan4`) to be configured and enter the Interface mode.
`awplus(config-if)#`	
`ip address 192.126.12.2/24`	Set the IP address on `vlan4`.

Names of commands used

dot1x port-control
radius-server host
radius-server key

Validation commands

show dot1x
show dot1x interface

Chapter 69: 802.1X Commands

Command List 69.2
debug dot1x 69.2
dot1x control-direction 69.3
dot1x eap 69.4
dot1x eapol-version 69.5
dot1x initialize interface 69.6
dot1x initialize supplicant 69.7
dot1x keytransmit 69.8
dot1x max-auth-fail 69.9
dot1x max-reauth-req 69.10
dot1x port-control 69.11
dot1x timeout tx-period 69.13
show debugging dot1x 69.14
show dot1x 69.15
show dot1x diagnostics 69.17
show dot1x interface 69.18
show dot1x sessionstatistics 69.23
show dot1x statistics interface 69.24
show dot1x supplicant 69.25
show dot1x supplicant interface 69.27
undebug dot1x 69.29

Command List

This chapter provides an alphabetical reference of commands used to configure 802.1X port access control.

debug dot1x

Use this command to enable 802.1X IEEE Port-Based Network Access Control troubleshooting functions.

Use the **no** variant of this command to disable this function.

Syntax

```
debug dot1x [all|auth-web|event|nsm|packet|timer]
no debug all dot1x
no debug dot1x [all|auth-web|event|nsm|packet|timer]
```

Parameter	Description
`all`	Used with the **no** variant of this command exclusively; turns off all debugging for 802.1X.
`auth-web`	Specifies debugging for 802.1X auth-web information.
`events`	Specifies debugging for 802.1X events.
`nsm`	Specifies debugging for NSM messages.
`packet`	Specifies debugging for 802.1X packets.
`timer`	Specifies debugging for 802.1X timers.

Mode Privileged Exec and Global Configuration

Usage This command without any parameters turns on normal 802.1X debug information.

```
awplus# debug dot1x
```

```
awplus# show debugging dot1x
```

```
802.1X debugging status:
  802.1X events debugging is
  802.1X timer debugging is on
  802.1X packets debugging is on
  802.1X NSM debugging is on
```

Examples

```
awplus# debug dot1x
awplus# debug dot1x all
```

Related Commands **show debugging dot1x**
undebug dot1x

dot1x control-direction

This command sets the direction of the filter for the unauthorized interface.

If the optional **in** parameter is specified with this command then packets entering the specified port are discarded. The **in** parameter discards the ingress packets received from the supplicant.

If the optional **both** parameter is specified with this command then packets entering (ingress) and leaving (egress) the specified port are discarded. The **both** parameter discards the packets received from the supplicant and sent to the supplicant.

The **no** variant of this command sets the direction of the filter to **both**. The port will then discard both ingress and egress traffic.

Syntax

```
dot1x control-direction {in|both}
no dot1x control-direction
```

Parameter	Description
`in`	Discard received packets from the supplicant (ingress packets).
`both`	Discard received packets from the supplicant (ingress packets) and transmitted packets to the supplicant (egress packets).

Default The authentication port direction is set to **both** by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Example s To set the port direction to the default (**both**) for `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x control-direction
```

To set the port direction to **in** for `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x control-direction in
```

Validation Commands

show dot1x
show dot1x interface
show auth-mac interface
show auth-web interface

dot1x eap

This command selects the transmit mode for the EAP packet. If the authentication feature is not enabled then EAP transmit mode is not enabled. The default setting discards EAP packets.

Syntax `dot1x eap {discard|forward|forward-untagged-vlan|forward-vlan}`

Parameter	Description
`discard`	Discard.
`forward`	Forward to all ports on the switch.
`forward-untagged-vlan`	Forward to ports with the same untagged VLAN.
`forward-vlan`	Forward to ports with the same VLAN.

Default The transmit mode is set to `discard` EAP packets by default.

Mode Global Configuration

Examples To set the transmit mode of EAP packet to `forward` to forward EAP packets to all ports on the switch, use the commands:

```
awplus# configure terminal
awplus(config)# dot1x eap forward
```

To set the transmit mode of EAP packet to `discard` to discard EAP packets, use the commands:

```
awplus# configure terminal
awplus(config)# dot1x eap discard
```

To set the transmit mode of EAP packet to `forward-untagged-vlan` to forward EAP packets to ports with the same untagged vlan, use the commands:

```
awplus# configure terminal
awplus(config)# dot1x eap forward-untagged-vlan
```

To set the transmit mode of EAP packet to `forward-vlan` to forward EAP packets to ports with the same vlan, use the commands:

```
awplus# configure terminal
awplus(config)# dot1x eap forward-vlan
```

dot1x eapol-version

This command sets the EAPOL protocol version for EAP packets when 802.1X port authentication is applied.

Use the **no** variant of this command to set the EAPOL protocol version to 1.

The default EAPOL protocol version is version 1.

Syntax

```
dot1x eapol-version {1|2}
no dot1x eapol-version
```

Parameter	Description
1	EAPOL version.
2	EAPOL version.

Default The EAP version for 802.1X authentication is set to 1 by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Examples To set the EAPOL protocol version to `2` for `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x eapol-version 2
```

To set the EAPOL protocol version to the default version (1) for interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x eapol-version
```

Validation Commands **show dot1x**
show dot1x interface

dot1x initialize interface

This command removes authorization for a connected **interface** with the specified *<interface-list>*. The connection will attempt to re-authorize when the specified **port** attempts to make use of the network connection.

Note Reauthentication could be a long time after the use of this command because the reauthorization attempt is not triggered by this command. The attempt is triggered by the first packet from the interface trying to access the network resources.

Syntax `dot1x initialize interface <interface-list>`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.6`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.2-1.0.4`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Examples To initialize 802.1X port authentication on the interface `port1.0.2`, use the command:

```
awplus# dot1x initialize interface port1.0.2
```

To unauthorize switch `port1.0.1` and attempt reauthentication on switch `port1.0.1`, use the command:

```
awplus# dot1x initialize interface port1.0.1
```

To unauthorize all switch ports for a 24-port device and attempt reauthentication, use the command:

```
awplus# dot1x initialize interface port1.0.1-port1.0.24
```

Validation Commands **show dot1x**
show dot1x interface

Related Commands **dot1x initialize supplicant**

dot1x initialize supplicant

This command removes authorization for a connected *supplicant* with the specified **MAC address** or **username**. The connection will attempt to re-authorize when the specified supplicant attempts to make use of the network connection.

Note Reauthentication could be a long time after the use of this command because the reauthorization attempt is not triggered by this command. The attempt is triggered by the first packet from the supplicant trying to access the network resources.

Syntax `dot1x initialize supplicant {<macadd>|username}`

Parameter	Description
`dot1x`	IEEE 802.1X Port-Based Access Control.
`initialize`	Initialize the port to attempt reauthentication.
`supplicant`	Specify the supplicant to initialize.
<macadd>	MAC (hardware address of the supplicant.
`username`	The name of the supplicant entry.

Mode Privileged Exec

Example To initialize the supplicant authentication, use the commands

```
awplus# configure terminal
awplus(config)# dot1x initialize supplicant
0090.99ab.a020
awplus(config)# dot1x initialize supplicant guest
```

Validation Commands **show dot1x**
show dot1x supplicant

Related Commands **dot1x initialize interface**

dot1x keytransmit

This command enables key transmission on the interface specified previously in Interface mode.

The **no** variant of this command disables key transmission on the interface specified.

Syntax

```
dot1x keytransmit
no dot1x keytransmit
```

Default Key transmission for port authentication is enabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to enable key transmission over an Extensible Authentication Protocol (EAP) packet between the authenticator and supplicant. Use the **no** variant of this command to disable key transmission.

Examples To enable the key transmit feature on interface `port1.0.2`, after it has been disabled by negation, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x keytransmit
```

To disable the key transmit feature from the default startup configuration on interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x keytransmit
```

Validation Commands **show dot1x**
show dot1x interface

dot1x max-auth-fail

Use this command to configure the maximum number of login attempts for a supplicant (client device) using the **auth-fail vlan** feature, when using 802.1X port authentication on an interface.

The **no** variant of this command resets the maximum login attempts for a supplicant (client device) using the auth-fail vlan feature, to the default configuration of 3 login attempts.

Syntax

```
dot1x max-auth-fail <0-10>
no dot1x max-auth-fail
```

Parameter	Description
<0-10>	Specify the maximum number of login attempts for supplicants on an interface using 802.1X port authentication.

Default The default maximum number of login attempts for a supplicant on an interface using 802.1X port authentication is three (3) login attempts.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage This command sets the maximum number of login attempts for supplicants on an interface. The supplicant is moved to the auth-fail VLAN from the Guest VLAN after the number of failed login attempts using 802.1X authentication is equal to the number set with this command.

See the related **auth auth-fail vlan** command on page 65.3. See also the section **"Failed Authentication VLAN" on page 64.33** for information about the auth-fail VLAN feature.

See the section **"Limitations on Allowed Feature Combinations" on page 64.33** for information about restrictions regarding combinations of authentication enhancements working together.

Examples To configure the maximum number of login attempts for a supplicant on interface `port1.0.2` to a single (`1`) login attempt, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x max-auth-fail 1
```

To configure the maximum number of login attempts for a supplicant on interface `port1.0.2` to the default number of three (`3`) login attempts, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x max-auth-fail
```

Validation Commands **show running-config**

Related Commands
auth auth-fail vlan
dot1x max-reauth-req
show dot1x interface

dot1x max-reauth-req

This command sets the number of reauthentication attempts before an interface is unauthorized.

The **no** variant of this command resets the reauthentication delay to the default.

Syntax

```
dot1x max-reauth-req <1-10>
no dot1x max-reauth-req
```

Parameter	Description
<1-10>	Specify the maximum number of reauthentication attempts for supplicants on an interface using 802.1X port authentication.

Default The default maximum reauthentication attempts for interfaces using 802.1X port authentication is two (2) reauthentication attempts, before an interface is unauthorized.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to set the maximum reauthentication attempts after failure.

Examples To configure the maximum number of reauthentication attempts for interface `port1.0.2` to a single (1) reauthentication request, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x max-reauth-req 1
```

To configure the maximum number of reauthentication attempts for interface `port1.0.2` to the default maximum number of two (2) reauthentication attempts, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x max-reauth-req
```

Validation Commands **show running-config**

Related Commands
dot1x max-auth-fail
show dot1x interface

dot1x port-control

This command enables 802.1X port authentication on the interface specified, and sets the control of the authentication port. When **port-control** is set to **auto**, the 802.1X authentication feature is executed on the interface, but only if the **aaa authentication dot1x** command has been issued.

The **no** variant of this command disables the port authentication on the interface specified.

Syntax

```
dot1x port-control {force-unauthorized|force-authorized|auto}
no dot1x port-control
```

Parameter	Description
`force-unauthorized`	Force port state to unauthorized. Specify to force a port to always be in an unauthorized state.
`force-authorized`	Force port state to authorized. Specify to force a port to always be in an authorized state.
`auto`	Allow port client to negotiate authentication. Specify to enable authentication on port.

Default 802.1X port control is disabled by default.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to force a port state. Note that all **dot1x** commands can only be applied to switch ports. They cannot be applied to dynamic (LACP) or static channel groups.

Examples To enable port authentication on the interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x port-control auto
```

To enable port authentication force authorized on the interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x port-control force-authorized
```

To disable port authentication on the interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x port-control
```

Validation Commands **show dot1x interface**

Related Commands **aaa authentication dot1x**

dot1x timeout tx-period

This command sets the transmit timeout for the authentication request on the specified interface.

The **no** variant of this command resets the transmit timeout period to the default (30 seconds).

Syntax

```
dot1x timeout tx-period <1-65535>
no dot1x timeout tx-period
```

Parameter	Description
<1-65535>	Seconds.

Default The default transmit period for port authentication is 30 seconds.

Mode Interface Configuration for a static channel, a dynamic (LACP) channel group, or a switch port.

Usage Use this command to set the interval between successive attempts to request an ID.

Examples To set the transmit timeout period to 5 seconds on interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# dot1x timeout tx-period 5
```

To reset transmit timeout period to the default (30 seconds) on interface `port1.0.2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no dot1x timeout tx-period
```

Validation Commands **show dot1x**
show dot1x interface

show debugging dot1x

Use this command to display the 802.1X debugging option set.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show debugging dot1x`

Mode User Exec and Privileged Exec

Usage This is a sample output from the `show debugging dot1x` command.

```
awplus# debug dot1x

awplus# show debugging dot1x

802.1X debugging status:
  802.1X events debugging is on
  802.1X timer debugging is on
  802.1X packets debugging is on
  802.1X NSM debugging is on
```

Example

```
awplus# show debugging dot1x
```

Related Commands **debug dot1x**

show dot1x

This command shows authentication information for dot1x (802.1X) port authentication.

If you specify the optional **all** parameter then this command also displays all authentication information for each port available on the switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dot1x [all]`

Parameter	Description
`all`	Displays all authentication information for each port available on the switch.

Mode Privileged Exec

Example

```
awplus# show dot1x all
```

Figure 69-1: Example output from the show dot1x command

```
awplus# show dot1x all
802.1X Port-Based Authentication Enabled
RADIUS server address: 150.87.18.89:1812
Next radius message id: 5
RADIUS client address: not configured
Authentication info for interface port1.0.6
portEnabled: true - portControl: Auto
portStatus: Authorized
reAuthenticate: disabled
reAuthPeriod: 3600
PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
PAE: connectTimeout: 30
BE: suppTimeout: 30 - serverTimeout: 30
CD: adminControlledDirections: in
KT: keyTxEnabled: false
critical: disabled
guestVlan: disabled
dynamicVlanCreation: single-dynamic-vlan
assignFailActionRule: deny
hostMode: multi-supplicant
maxSupplicant: 1024
dot1x: enabled
protocolVersion: 1
authMac: enabled
method: PAP
reauthRelearning: disabled
authWeb: enabled
method: PAP
lockCount: 3
packetForwarding: disabled
twoStepAuthentication:
    configured: enabled
    actual: enabled
SupplicantMac: none
```

Figure 69-1: Example output from the show dot1x command (cont.)

```
supplicantMac: none
Supplicant name: manager
Supplicant address: 00d0.59ab.7037
    authenticationMethod: 802.1X Authentication
    portStatus: Authorized - currentId: 1
    abort:F fail:F start:F timeout:F success:T
    PAE: state: Authenticated - portMode: Auto
    PAE: reAuthCount: 0 - rxRespId: 0
    PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
    BE: state: Idle - reqCount: 0 - idFromServer: 0
    CD: adminControlledDirections: in - operControlledDirections: in
    CD: bridgeDetected: false
    KR: rxKey: false
    KT: keyAvailable: false - keyTxEnabled: false
    criticalState: off
    dynamicVlanId: 2
802.1X statistics for interface port1.0.6
    EAPOL Frames Rx: 5 - EAPOL Frames Tx: 16
    EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
    EAP Rsp/Id Frames Rx: 3 - EAP Response Frames Rx: 2
    EAP Req/Id Frames Tx: 8 - EAP Request Frames Tx: 2
    Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
    EAPOL Last Frame Version Rx: 1 - EAPOL Last Frame Src:  00d0.59ab.7037
Authentication session statistics for interface port1.0.6
    session user name: manager
    session authentication method: Remote server
    session time: 19440 secs
    session terminate cause: Not terminated yet
Authentication Diagnostics for interface port1.0.6
    Supplicant address: 00d0.59ab.7037
    authEnterConnecting: 2
    authEaplogoffWhileConnecting: 1
    authEnterAuthenticating: 2
    authSuccessWhileAuthenticating: 1
    authTimeoutWhileAuthenticating: 1
    authFailWhileAuthenticating: 0
    authEapstartWhileAuthenticating: 0
    authEaplogoggWhileAuthenticating: 0
    authReauthsWhileAuthenticated: 0
    authEapstartWhileAuthenticated: 0
    authEaplogoffWhileAuthenticated: 0
    BackendResponses: 2
    BackendAccessChallenges: 1
    BackendOtherrequestToSupplicant: 3
    BackendAuthSuccess: 1
    BackendAuthFails: 0
```

show dot1x diagnostics

This command shows 802.1X authentication diagnostics for the specified interface (optional), which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

If no interface is specified then authentication diagnostics are shown for all interfaces.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dot1x diagnostics [interface <interface-list>]`

Parameter	Description
`interface`	Specify a port to show.
`<interface-list>`	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.2-1.0.4`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example See the sample output below showing 802.1X authentication diagnostics for `port1.0.5`:

```
awplus# show dot1x diagnostics interface port1.0.5
```

Output **Figure 69-2: Example output from the show dot1x diagnostics command**

```
Authentication Diagnostics for interface port1.0.5
   Supplicant address: 00d0.59ab.7037
     authEnterConnecting: 2
     authEaplogoffWhileConnecting: 1
     authEnterAuthenticating: 2
     authSuccessWhileAuthenticating: 1
     authTimeoutWhileAuthenticating: 1
     authFailWhileAuthenticating: 0
     authEapstartWhileAuthenticating: 0
     authEaplogoggWhileAuthenticating: 0
     authReauthsWhileAuthenticated: 0
     authEapstartWhileAuthenticated: 0
     authEaplogoffWhileAuthenticated: 0
     BackendResponses: 2
     BackendAccessChallenges: 1
     BackendOtherrequestToSupplicant: 3
     BackendAuthSuccess: 1
```

show dot1x interface

This command shows the status of 802.1X port-based authentication on the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

Use the optional **diagnostics** parameter to show authentication diagnostics for the specified interfaces. Use the optional **sessionstatistics** parameter to show authentication session statistics for the specified interfaces. Use the optional **statistics** parameter to show authentication diagnostics for the specified interfaces. Use the optional **supplicant** parameter to show the supplicant state for the specified interfaces.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show dot1x interface <interface-list> [diagnostics|sessionstatistics|
   statistics|supplicant [brief]]
```

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.3-1.0.5`. Do not mix interface types in a list The specified interfaces must exist.
`diagnostics`	Diagnostics.
`sessionstatistics`	Session Statistics.
`statistics`	Statistics.
`supplicant`	Supplicant.
`brief`	Brief summary of supplicant state.

Mode Privileged Exec

Examples See the sample output below showing 802.1X authentication status for `port1.0.6`:

```
awplus# show dot1x interface port1.0.6
```

Figure 69-3: Example output from the show dot1x interface command for a port

```
awplus#show dot1x interface port1.0.6
Authentication info for interface port1.0.6
    portEnabled: true - portControl: Auto
    portStatus: Authorized
    reAuthenticate: disabled
    reAuthPeriod: 3600
    PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
    PAE: connectTimeout: 30
    BE: suppTimeout: 30 - serverTimeout: 30
    CD: adminControlledDirections: in
    KT: keyTxEnabled: false
    critical: disabled
    guestVlan: disabled
    dynamicVlanCreation: single-dynamic-vlan
        assignFailActionRule: deny
    hostMode: multi-supplicant
        maxSupplicant: 1024
    dot1x: enabled
        protocolVersion: 1
    authMac: enabled
        method: PAP
        reauthRelearning: disabled
    authWeb: enabled
        method: PAP
        lockCount: 3
        packetForwarding: disabled
     twoStepAuthentication:
        configured: enabled
        actual: enabled
supplicantMac: none
```

See the sample output below showing 802.1X authentication `sessionstatistics` for `port1.0.6`:

```
awplus# show dot1x interface port1.0.6 sessionstatistics
```

```
awplus#show dot1x interface port1.0.6 sessionstatistics
Authentication session statistics for interface port1.0.6
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

See sample output below showing 802.1X authentication `diagnostics` for `port1.0.6`:

```
awplus# show dot1x interface port1.0.6 diagnostics
```

```
awplus#show dot1x interface port1.0.6 diagnostics
Authentication Diagnostics for interface port1.0.6
    Supplicant address: 00d0.59ab.7037
        authEnterConnecting: 2
        authEaplogoffWhileConnecting: 1
        authEnterAuthenticating: 2
        authSuccessWhileAuthenticating: 1
        authTimeoutWhileAuthenticating: 1
        authFailWhileAuthenticating: 0
        authEapstartWhileAuthenticating: 0
        authEaplogoggWhileAuthenticating: 0
        authReauthsWhileAuthenticated: 0
        authEapstartWhileAuthenticated: 0
        authEaplogoffWhileAuthenticated: 0
        BackendResponses: 2
        BackendAccessChallenges: 1
        BackendOtherrequestToSupplicant: 3
        BackendAuthSuccess: 1
```

See sample output below showing the `supplicant` on the interface `port1.0.6`:

```
awplus# show dot1x interface port1.0.6 supplicant
```

```
awplus#show dot1x interface port1.0.6 supplicant
authenticationMethod: dot1x
    totalSupplicantNum: 1
    authorizedSupplicantNum: 1
        macBasedAuthenticationSupplicantNum: 0
        dot1xAuthenticationSupplicantNum: 1
        webBasedAuthenticationSupplicantNum: 0
Supplicant name: manager
Supplicant address: 00d0.59ab.7037
    authenticationMethod: dot1x
    portStatus: Authorized - currentId: 4
    abort:F fail:F start:F timeout:F success:T
    PAE: state: Authenticated - portMode: Auto
    PAE: reAuthCount: 0 - rxRespId: 0
    PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
    BE: state: Idle - reqCount: 0 - idFromServer: 3
    BE: suppTimeout: 30 - serverTimeout: 30
    CD: adminControlledDirections: in - operControlledDirections:
in
    CD: bridgeDetected: false
    KR: rxKey: false
    KT: keyAvailable: false - keyTxEnabled: false
```

See sample output below showing 802.1X (`dot1x`) authentication statistics for `port1.0.6`:

```
awplus# show dot1x statistics interface port1.0.6
```

```
awplus#show dot1x statistics interface port1.0.6
802.1X statistics for interface port1.0.6
    EAPOL Frames Rx: 5 - EAPOL Frames Tx: 16
    EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
    EAP Rsp/Id Frames Rx: 3 - EAP Response Frames Rx: 2
    EAP Req/Id Frames Tx: 8 - EAP Request Frames Tx: 2
    Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
    EAPOL Last Frame Version Rx: 1 - EAPOL Last Frame
Src:00d0.59ab.7037
```

Table 69-1: Parameters in the output of the show dot1x interface command

Parameter	Description
`portEnabled`	Interface operational status (Up-true/down-false).
`portControl`	Current control status of the port for 802.1X control.
`portStatus`	802.1X status of the port (authorized/unauthorized).
`reAuthenticate`	Reauthentication enabled/disabled status on port.
`reAuthPeriod`	Value holds meaning only if reauthentication is enabled.
`abort`	Indicates that authentication should be aborted when set to true.
`fail`	Indicates failed authentication attempt when set to false.
`start`	Indicates authentication should be started when set to true.
`timeout`	Indicates authentication attempt timed out when set to true.
`success`	Indicates authentication successful when set to true.
`state`	Current 802.1X operational state of interface.
`mode`	Configured 802.1X mode.
`reAuthCount`	Reauthentication count.
`quietperiod`	Time between reauthentication attempts.
`reAuthMax`	Maximum reauthentication attempts.
`BE`	Backend authentication state machine variables and constants.
`state`	State of the state machine.
`reqCount`	Count of requests sent to server.
`suppTimeout`	Supplicant timeout.
`serverTimeout`	Server timeout.
`maxReq`	Maximum requests to be sent.
`CD`	Controlled Directions State machine.

Table 69-1: Parameters in the output of the show dot1x interface command (cont.)

Parameter	Description
`adminControlledDirections`	Administrative value (Both/In).
`operControlledDirections`	Operational Value (Both/In).
`KR`	Key receive state machine.
`rxKey`	True when EAPOL-Key message is received by supplicant or authenticator. false when key is transmitted.
`KT`	Ket Transmit State machine.
`keyAvailable`	False when key has been transmitted by authenticator, true when new key is available for key exchange.
`keyTxEnabled`	Key transmission enabled/disabled status.

Related Commands

show auth-web diagnostics
show dot1x sessionstatistics
show dot1x statistics interface
show dot1x supplicant interface

show dot1x sessionstatistics

This command shows authentication session statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dot1x sessionstatistics [interface <interface-list>]`

Parameter	Description
`interface`	Specify a port to show.
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example See sample output below showing 802.1X (`dot1x`) authentication session statistics for `port1.0.6`:

```
awplus# show dot1x sessionstatistics interface port1.0.6
```

```
Authentication session statistics for interface port1.0.6
    session user name: manager
        session authentication method: Remote server
        session time: 19440 secs
        session terminat cause: Not terminated yet
```

show dot1x statistics interface

This command shows the authentication statistics for the specified interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dot1x statistics interface <interface-list>`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.

Mode Privileged Exec

Example See sample output below showing 802.1X authentication statistics for `port1.0.6`:

```
awplus# show dot1x statistics interface port1.0.6
```

```
802.1X statistics for interface
    EAPOL Frames Rx: 5 - EAPOL Frames Tx: 16
    EAPOL Start Frames Rx: 0 - EAPOL Logoff Frames Rx: 0
    EAP Rsp/Id Frames Rx: 3 - EAP Response Frames Rx: 2
    EAP Req/Id Frames Tx: 8 - EAP Request Frames Tx: 2
    Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
    EAPOL Last Frame Version Rx: 1 - EAPOL Last Frame
Src:00d0.59ab.7037
```

show dot1x supplicant

This command shows the supplicant state of the authentication mode set for the switch.

This command shows a summary when the optional **brief** parameter is used.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dot1x supplicant [<macadd>] [brief]`

Parameter	Description
<macadd>	MAC (hardware) address of the Supplicant.
`brief`	Brief summary of the Supplicant state.

Mode Privileged Exec

Example See sample output below showing the 802.1X authenticated supplicant on the switch:

```
awplus# show dot1x supplicant
```

```
authenticationMethod: dot1x
totalSupplicantNum: 1
authorizedSupplicantNum: 1
macBasedAuthenticationSupplicantNum: 0
dot1xAuthenticationSupplicantNum: 1
webBasedAuthenticationSupplicantNum: 0
Supplicant name: manager
Supplicant address: 00d0.59ab.7037
    authenticationMethod: dot1x
    Two-Step Authentication:
      firstAuthentication: Pass - Method: mac
      secondAuthentication: Pass - Method: dot1x
portStatus: Authorized - currentId: 4
abort:F fail:F start:F timeout:F success:T
PAE: state: Authenticated - portMode: Auto
PAE: reAuthCount: 0 - rxRespId: 0
PAE: quietPeriod: 60 - maxReauthReq: 2 - txPeriod: 30
BE: state: Idle - reqCount: 0 - idFromServer: 3
BE: suppTimeout: 30 - serverTimeout: 30
CD: adminControlledDirections: in - operControlledDirections: in
CD: bridgeDetected: false
KR: rxKey: false
KT: keyAvailable: false - keyTxEnabled: false
```

See sample output below showing the supplicant on the switch using the `brief` parameter:

```
awplus# show dot1x supplicant 00d0.59ab.7037 brief
```

```
Interface port1.0.6
   authenticationMethod: dot1x
   totalSupplicantNum: 1
   authorizedSupplicantNum: 1
       macBasedAuthenticationSupplicantNum: 0
       dot1xAuthenticationSupplicantNum: 1
       webBasedAuthenticationSupplicantNum: 0
Interface   VID Mode MAC Address     Status        IP Address     Username
=========   === ==== ===========     ======        ==========     ========
port1.0.6   2   D    00d0.59ab.7037  Authenticated 192.168.2.201  manager
```

See sample output below showing the supplicant on the switch using the `brief` parameter:

```
awplus# show dot1x supplicant brief
```

For example, if two-step authentication is configured with 802.1X authentication as the first method and web authentication as the second method then the output is as follows:

```
Interface port1.0.6
  authenticationMethod: dot1x/web
  Two-Step Authentication
    firstMethod: dot1x
    secondMethod: web
  totalSupplicantNum: 1
  authorizedSupplicantNum: 1
    macBasedAuthenticationSupplicantNum: 0
    dot1xAuthenticationSupplicantNum: 0
    webBasedAuthenticationSupplicantNum: 1
    otherAuthenticationSupplicantNum: 0

Interface   VID  Mode MAC Address    Status            IP Address      Username
=========== ==== ==== ============== ================= =============== ========
port1.0.6   5    W    0008.0d5e.c216 Authenticated     192.168.1.200   web
```

Related Commands **show dot1x supplicant interface**

show dot1x supplicant interface

This command shows the supplicant state of the authentication mode set for the interface, which may be a static channel (or static aggregator) or a dynamic (or LACP) channel group or a switch port.

This command shows a summary when the optional **brief** parameter is used.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dot1x supplicant interface <interface-list> [brief]`

Parameter	Description
<interface-list>	The interfaces or ports to configure. An interface-list can be: ■ an interface (e.g. `vlan2`), a switch port (e.g. `port1.0.6`), a static channel group (e.g. `sa2`) or a dynamic (LACP) channel group (e.g. `po2`) ■ a continuous range of interfaces, ports, static channel groups or dynamic (LACP) channel groups separated by a hyphen; e.g. `vlan2-8`, or `port1.0.1-1.0.4`, or `sa1-2`, or `po1-2` ■ a comma-separated list of the above; e.g. `port1.0.1,port1.0.4-1.0.6`. Do not mix interface types in a list The specified interfaces must exist.
`brief`	Brief summary of the Supplicant state.

Mode Privileged Exec

Examples See sample output below showing the supplicant on the interface `port1.0.6`:

```
awplus# show dot1x interface port1.0.6
```

```
Interface port1.0.6
 authenticationMethod: dot1x
  totalSupplicantNum: 1
 authorizedSupplicantNum: 1
   macBasedAuthenticationSupplicantNum: 0
   dot1xAuthenticationSupplicantNum: 1
   webBasedAuthenticationSupplicantNum: 0
   otherAuthenticationSupplicantNum: 0

 Supplicant name: VCSPCVLAN10
 Supplicant address: 0000.cd07.7b60
   authenticationMethod: 802.1X
 Two-Step Authentication:
   firstAuthentication: Pass - Method: mac
   secondAuthentication: Pass - Method: dot1x
   portStatus: Authorized - currentId: 3
   abort:F fail:F start:F timeout:F success:T
   PAE: state: Authenticated - portMode: Auto
   PAE: reAuthCount: 0 - rxRespId: 0
   PAE: quietPeriod: 60 - maxReauthReq: 2
   BE: state: Idle - reqCount: 0 - idFromServer: 2
   CD: adminControlledDirections:in -
operControlledDirections:in
   CD: bridgeDetected: false
   KR: rxKey: false
   KT: keyAvailable: false - keyTxEnabled: false
```

See sample output below showing the supplicant on the switch using the `brief` parameter:

```
awplus# show dot1x supplicant interface brief
```

```
Interface port1.0.6
    authenticationMethod: dot1x
Two-Step Authentication:
   firstMethod: mac
   secondMethod: dot1x
totalSupplicantNum: 1
authorizedSupplicantNum: 1
macBasedAuthenticationSupplicantNum: 0
dot1xAuthenticationSupplicantNum: 1
webBasedAuthenticationSupplicantNum: 0

Interface   VID  Mode MAC Address     Status          IP Address      Username
=========   ===  ==== ===========     ======          ==========      ========
port1.0.6   2    D    00d0.59ab.7037  Authenticated   192.168.2.201   manager
```

See the sample output below for static channel group (static aggregator) interface `sa1`:

```
awplus# show dot1x interface sa1 supplicant brief
```

```
awplus#show dot1x interface sa1 supplicant brief
Interface sa1
  authenticationMethod: dot1x
Two-Step Authentication:
   firstMethod: mac
   secondMethod: dot1x
  totalSupplicantNum: 1
  authorizedSupplicantNum: 1
    macBasedAuthenticationSupplicantNum: 0
     dot1xAuthenticationSupplicantNum: 1
     webBasedAuthenticationSupplicantNum: 0
     otherAuthenticationSupplicantNum: 0

Interface   VID  Mode MAC Address    Status            IP Address      Username
=========   ==== ==== ============== ================= =============== ========
sa1         1    D    00d0.59ab.7037 Authenticated     --              test1
```

Related Commands **show dot1x supplicant**

undebug dot1x

This command applies the functionality of the **no debug dot1x** command on page 69.2.

Chapter 70: RADIUS Introduction and Configuration

Introduction 70.2
RADIUS Packets 70.3
RADIUS Attributes 70.4
RADIUS Security 70.5
RADIUS Proxy 70.6
RADIUS Accounting 70.7
RADIUS Configuration 70.9
Switch Configuration Tasks 70.9
Switch to RADIUS Server Communication 70.10
AAA Server Groups Configuration 70.12
RADIUS Configuration Examples 70.15
RADIUS Authentication 70.15
Single RADIUS Server Configuration 70.16
Multiple RADIUS Server Configuration 70.17
RADIUS Server Group Configuration 70.17
RADIUS Server Configuration using Server Groups 70.18

Introduction

The main purpose of RADIUS (Remote Authentication Dial In User Service) is to enable the authentication of network users stored in a database on a server known as a RADIUS Server.

When users connect to the network, the switch the users connect to can challenge the users for authentication, and pass on the authentication to the RADIUS server to check. Based on the result of the check against the database, the RADIUS Server informs the switch whether or not to allow the connected user access to the network.

A RADIUS Server can do more than allow or deny access to the network. A RADIUS Server can send back parameters to the connected users, such as an IP address for the user, or a VLAN for the user, or a privilege level for a session. RADIUS also provides an accounting service. Switches can inform the RADIUS Server how long a user has been connected to the network, and how much traffic the user has sent and received while connected to the network.

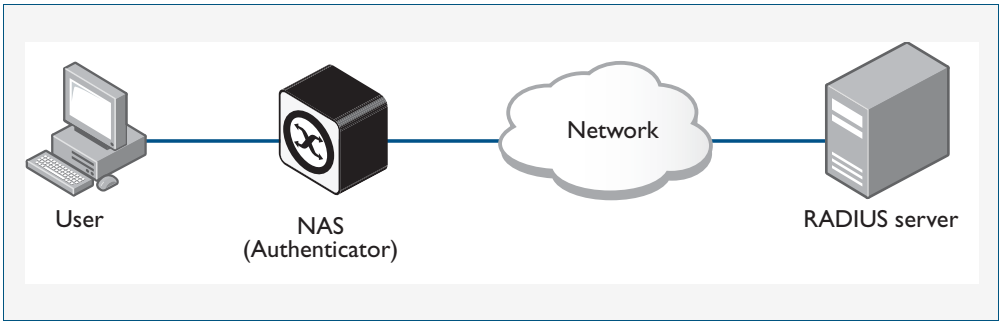
The original use for RADIUS was for the authentication of users dialing into an ISP (Internet Service Provider). A PPP (Point-to-Point Protocol) connection would be established between the remote client and the ISP's access switch. The ISP's access switch would receive the client's username and password using PAP (Password Authentication Protocol) or using CHAP (Challenge Handshake Authentication Protocol) and pass on the client's username and password to the RADIUS server to authenticate the client. The RADIUS Server's response to the authentication request would be sent back to the client as a PAP or CHAP allow or deny.

RADIUS has been adapted to network access authentication applications. Network access authentication using RADIUS follows a similar method to the PPP dial-up application for ISPs. For general network access authentication there is the RADIUS Server where the database of user authentication data is stored and a NAS (Network Access Server), which is the switch that user connects to first. The RADIUS Server and the NAS communicate with each other through exchanging attributes. Usernames and passwords are treated as attributes in RADIUS packets to and from a RADIUS Server and a NAS. The RADIUS Server is configured with a list of valid NASs that are allowed to send authentication requests to the RADIUS Server.

The RADIUS Server will not accept authentication requests from a NAS that is not on the list of valid NASs. Each NAS has a shared secret, which is a shared key with the RADIUS Server that is used to authenticate requests. The RADIUS Server has access to a list of user authentication data, stored within the RADIUS Server or accessed from another server.

Communication between the NAS and RADIUS Server uses the RADIUS protocol. The RADIUS protocol uses UDP packets. There are two UDP ports used as the destination port for RADIUS authentication packets (ports 1645 and 1812). Note that port 1812 is in more common use than port 1645 for authentication packets. UDP ports (ports 1646 and 1813) are used for RADIUS accounting separately from the ports used for RADIUS authentication.

Figure 70-1: Example showing a user to a NAS to a RADIUS server network connection



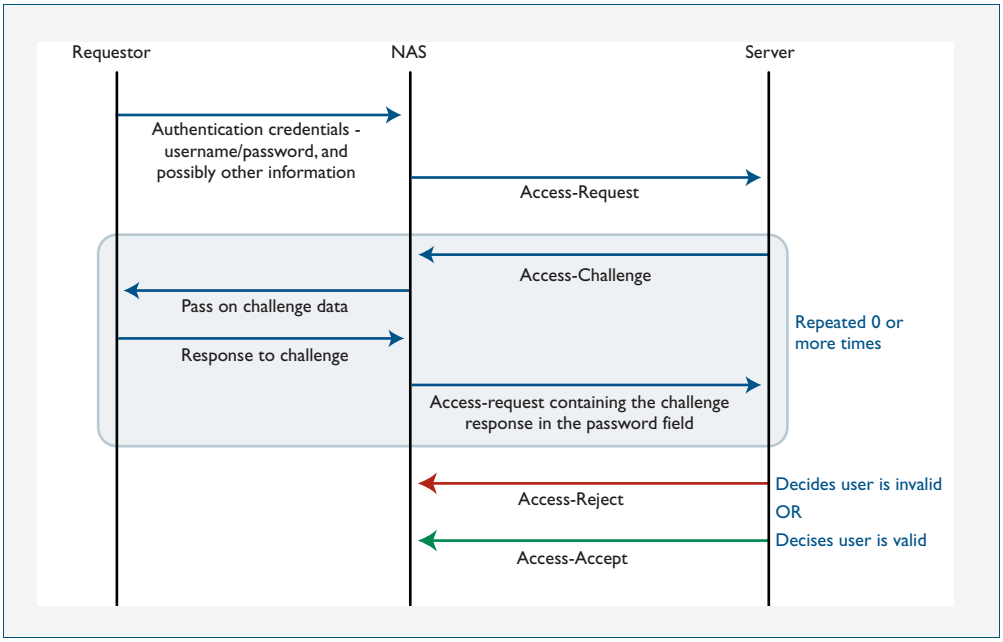
RADIUS Packets

The RADIUS RFCs define the RADIUS packet types and attributes. RADIUS authentication is defined by RFC2058, RFC2138, RFC2865, and RFC2868. RADIUS accounting is defined by RFC2059, RFC2139, RFC2866, and RFC2867. These RADIUS RFCs define over fifty attributes and six packets types (`Access-Request, Access-Accept, Access-Reject, Accounting-Request, Accounting-Response, Access-Challenge`).

A RADIUS exchange is initiated by the NAS when a user requests access to the NAS. The NAS obtains the user authentication data adds them into a RADIUS `Access-Request` packet type and sends the RADIUS `Access-Request` packet to the RADIUS Server.

- If a RADIUS Server has not been configured for authentication request from a NAS then it will silently discard an `Access-Request` packet from it.
- If the RADIUS Server accepts the request from the NAS it considers the authentication date provided in the `Access-Request` packet. The RADIUS Server may verify the user from its own database or it may connect to other servers to verify.
- If the RADIUS Server decides that the user is not allowed access to the NAS it responds to the NAS with an `Access-Reject` packet and the NAS will block the user.
- If the RADIUS Server decides that the user is valid but needs more information to verify that the user is not an impostor, it may send an `Access-Challenge` packet to the NAS that the NAS forwards to the user. The NAS forwards the user response to the `Access-Challenge` packet in an `Access-Request` packet to the RADIUS Server to accept or reject to allow or deny NAS user access.
- If the RADIUS Server rejects the user it sends an `Access-Reject` packet to the NAS.
- If the RADIUS Server accepts the user it sends an `Accept-Accept` packet to the NAS. The `Accept-Accept` packet to the NAS contains attributes that the NAS can apply.

Figure 70-2: Example showing an exchange from a requestor to a NAS to a RADIUS server



RADIUS Attributes

Each attribute is identified by its RFC-defined name, followed by its attribute ID in parenthesis.

- **User-name(1)**

  User-names are strings of at least three characters and have a maximum of 253 characters, which is the upper limit on all RADIUS attributes.

- **User-password(2)**

  User-passwords are encrypted using an MD5 hash of the password, the NAS's shared secret with the RADIUS Server, and a request authenticator value. User-passwords can either be used at the initial authentication attempt or in response to an Access-Challenge packet type from the RADIUS Server to the NAS.

- **CHAP-password(3)**

  CHAP-passwords are used if the NAS is using CHAP to authenticate the user, and doesn't receive the use the user's password but sends the CHAP response to the RADIUS Server instead. The CHAP password is an encrypted string that is an MD5 hash of the password and challenge value sent by the user.

- **Framed-IP-Address(8)**

  Used for dial-in user making PPP connections to the NAS who are dynamically allocated an IP address that they can use for the duration of their connect. The RADIUS Server sends the Framed-IP-Address to the NAS to allocate.

- **Service-Type(6)**

  Used when the NAS is authenticating a user who wants to open a management session on the NAS, and is sent by the RADIUS Server back to the NAS in an Access-Accept type packet to indicate the level of access the NAS gives a user. Service-Type(6) is mapped to a Privileged management session for AlliedWare Plus.

- **NAS-Port-Type(61)**

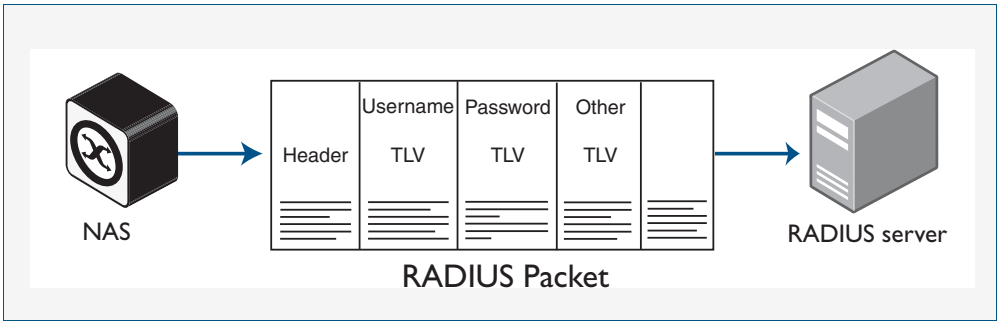
Identifies the type of port on which the user is accessing the NAS. The NAS-Port-Type(61) attribute is sent by the NAS to the RADIUS Server in Access-Request type packet, so the RADIUS Server may use it to choose access type. For 802.1X sessions, the NAS-Port-Type sent by the NAS is Ethernet (15).

- **802.1X VLAN assignment uses:**

  Tunnel-Type(64), Tunnel-Medium-Type(65), Tunnel-Private-Group-ID(81), Egress-VLANID(56), and Egress-VLAN-Name(58) attributes (specified in RFC4675 used to specify 802.1Q tagged and untagged VLAN assignments with LLDP-MED/Voice-VLAN).

Attributes are carried within RADIUS packets in the form of TLVs (Type Length Values). Every attribute has an attribute ID number in the Type field of the TLV. The Length field holds a one-byte number that represents then length of the TLV. The Value field holds the value of the attribute.

Figure 70-3: Example showing TLVs in a RADIUS packet from a NAS to a RADIUS server



RADIUS Security

RADIUS is used for network security and carries user authentication information, so can be a target for security attacks. To counter threats there are three elements to RADIUS security:

- **Shared secret**
- **Authenticator**
- **Password Encryption**

Shared secret

Every NAS and server are configured with a pre-shared key, called the “shared secret”, which is a key string, with no particular format of at least 16 characters.

The protocol has no method for choosing and sharing the secret between the NAS and the server. The secret must be manually generated and separately configured on the NAS and on the server.

The shared secret itself never appears in any RADIUS packets. It is used as an input to the algorithms used for creating encrypted values that are carried in the packets.

Authenticator

The authenticator is a random 16-byte value generated by the NAS. The NAS creates a new authenticator value for each `Access-Request` that it sends.

The response packets that come back from the server contain a value called the Response Authenticator. This is a value that is created by performing an MD5 hash on a string that is created by concatenating the packet type identifier, Session ID, Authenticator sent in the request packet, Attribute fields in the packet, Shared secret that the server shares with the NAS to which it is responding.

When the NAS receives the response packet, it performs the same hash on the same values, and verifies that it comes up with the same result. If not, then it must assume that the response packet has been spoofed, and silently discards it.

Password encryption

The value placed in the user-password TLV of an `Access-Request` packet is not simply an exact copy of the password sent from the requestor to the NAS.

The NAS concatenates together the shared secret and the authenticator that it has randomly generated for this request and then performs manipulations (MD5, XOR) on that concatenation, and the password to create the value to go into password TLV.

When the server validates the `Access-Request`, it retrieves the user's password from the user credentials database, and performs the same manipulation upon that password. If the result matches the value in the user-password field of the `Access-Request`, then the password sent by the requestor is deemed to be correct.

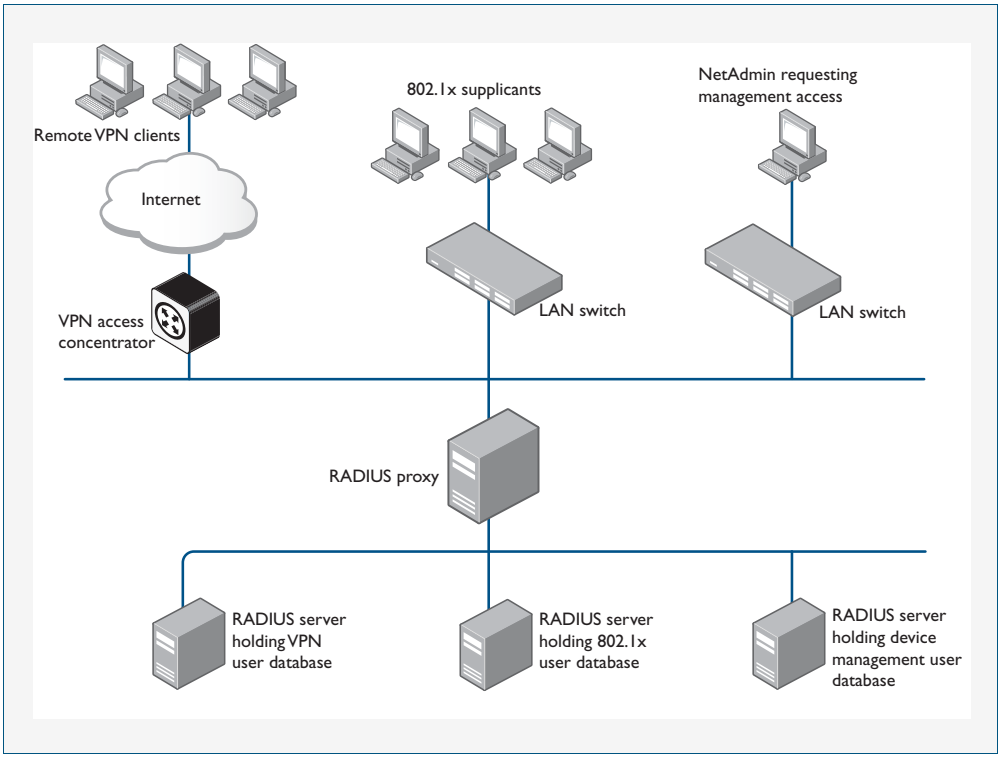
RADIUS Proxy

The user database, which user credentials sent to a RADIUS server are looked up in, may not reside on the RADIUS server itself. The external user database may reside on another RADIUS server, and the communication to that server uses RADIUS. In the case where a RADIUS server communicates with a NAS, but also acts as a client to another RADIUS server, is said to be acting as a RADIUS proxy.

There are a variety of situations where RADIUS proxy is useful. Multiple RADIUS servers could have been set up, holding user databases for different purposes such as Authentication, Switch management sessions, Authenticating VPN connections, and Authenticating 802.1X sessions.

But it is convenient for there to be just one address that all the NASs in the network use as their RADIUS server. That one RADIUS server that the NASs send their requests to, can act as a proxy for all the servers holding the different user databases.

Figure 70-4: Example showing RADIUS proxy



RADIUS Accounting

There are only two types of RADIUS accounting packet: `Accounting-Request` and `Accounting-Response`.

The `Accounting-Request` packets are always sent from the NAS to the server. The `Accounting-Response` packets are always sent from the server to the NAS, and are effectively ACKs of the `Accounting-Request` packets.

The `Accounting-Request` packets always carry the attribute `Acct-Status-Type`. The most commonly used values of this attribute are:

- **Start** – which denotes a packet marking that a session is beginning
- **Stop** – which denotes a packet marking that a session is ending
- **Interim update** – packets sent periodically during the session to give update reports on the statistics that are being collected.

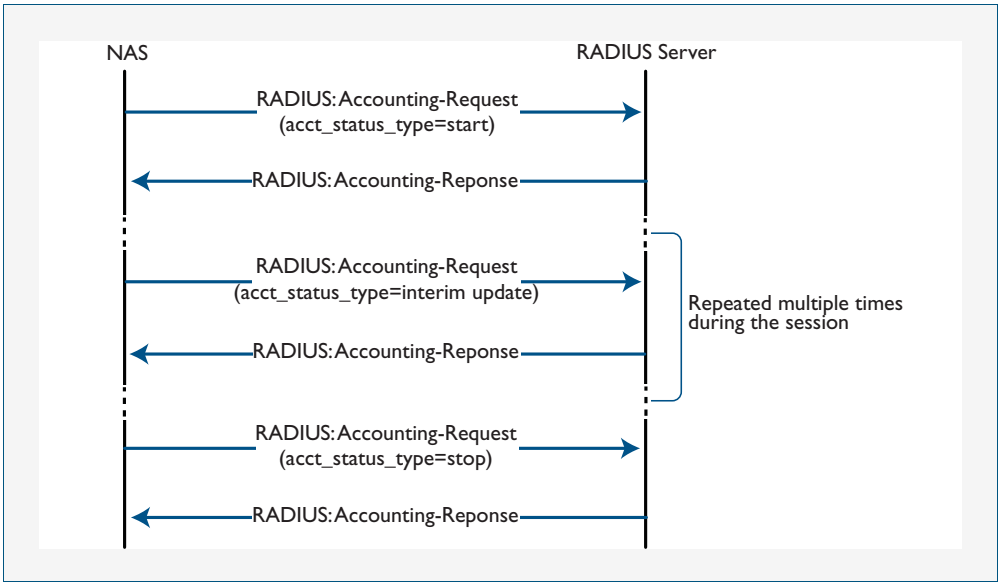
The statistics that can be exchanged in the session are:

- **Input Octets**
- **Input Packets**
- **Output Octets**
- **Output Packets**
- **Session Duration**

There is no requirement to exchange all these statistics – NAS implementations are at liberty to choose which statistics they will send. Each of these statistics has a corresponding attribute type. The attributes are sent in Interim-Update and Stop accounting request packets.

Each accounting session has a unique session ID, which is chosen by the NAS. The session ID is carried in an `Acct-Session-Id` attribute, that should be present in every packet involved in the session. The accounting packets typically do not use the same UDP port as the authentication packets. The default port for RADIUS accounting is 1813.

Figure 70-5: Example showing RADIUS accounting between a NAS and a RADIUS server



RADIUS Configuration

This section describes how to configure RADIUS with the available AAA commands. For a description of AAA commands, refer to the **AAA Commands** chapter. For a description of the RADIUS commands used, refer to the **RADIUS Commands** chapter.

RADIUS is often used in a variety of networks that need high security while maintaining access for remote users. RADIUS is suitable for the following networks that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database.
- Networks in which a user may access a single service. Using RADIUS, you can control user access to a single host, or to a single utility such as Telnet.
- Networks that require accounting. You can use RADIUS accounting independent of RADIUS authentication. The RADIUS accounting functions allow data to be sent at the start and end of services, indicating the amount of resources (time, packets, bytes) used.

Switch Configuration Tasks

To configure RADIUS on your switch or access server, you must perform the following tasks:

- Use the **aaa authentication** command to define method lists for RADIUS authentication. For information about this command, refer to the **AAA Commands** chapter.
- Use authentication commands to enable the defined method lists to be used. For more information, refer to the **Authentication Commands** chapter.

The following configuration tasks are optional:

- You can use the **aaa group server** command to group selected RADIUS hosts for specific services. For detailed information about this command, refer to the **AAA Server Groups Configuration** section in this chapter and refer to the **AAA Commands** chapter.
- You can use the **aaa accounting login** command to enable accounting for RADIUS connections. For information about this command, refer to the **AAA Commands** chapter.

This section describes how to set up RADIUS for authentication and accounting on your network, and includes the following sections:

- Switch to RADIUS Server Communication (Required)
- Configuring AAA Server Groups (Optional)
- Configuring AAA Server Groups with Deadtime (Optional)
- Specifying RADIUS Authentication
- Specifying RADIUS Accounting (Optional)

For RADIUS configuration examples using the commands in this chapter, refer to the section **RADIUS Configuration Examples** at the end of this chapter.

Switch to RADIUS Server Communication

The RADIUS host is normally a multiuser system running RADIUS server software from a software provider. Switch to RADIUS server communication has several components:

- Host name or IP address
- Authentication destination port
- Accounting destination port
- Timeout period
- Retransmission value
- Key string

RADIUS security servers are identified on the basis of their host name or IP address, host name and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

A RADIUS server and a switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS using the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text string that it shares with the switch, which you can specify using the **key** parameter in the **radius-server host** command.

The timeout, retransmission, and encryption key values are configurable globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings. To apply these settings globally to all RADIUS servers communicating with the switch, use the three global commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**. To apply these values on a specific RADIUS server, use the **radius-server host** command.

Note

You can configure both global and per-server timeout, retransmission, and key value commands simultaneously on the same Network Access Server.

If both global and per-server functions are configured on a switch, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands.

To configure per-server RADIUS server communication, use the following command in the Global Configuration mode:

Mode and Command	Command Purpose
awplus(config)# radius-server host {<*hostname*>| <*ip-address*>} [auth-port <*port-number*>] [acct-port <*port-number*>] [timeout <*seconds*>] [retransmit <*retries*>] [key <*string*>]	Specifies the IP address or host name of the remote RADIUS server host and assigns authentication and accounting destination UDP port numbers. Use the `auth-port <port-number>` option to configure a specific UDP port on this RADIUS server to be used solely for authentication. Use the `acct-port <port-number>` option to configure a specific UDP port on this RADIUS server to be used solely for accounting. To configure the network access server to recognize more than one host entry associated with a single IP address, simply repeat this command as many times as necessary, making sure that each UDP port number is different. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host. If no timeout is set, the global value is used; otherwise, enter a value in the range 1 to 1000. If no retransmit value is set, the global value is used; otherwise enter a value in the range 1 to 1000. If no key string is specified, the global value is used.

To configure global communication settings between the switch and a RADIUS server, use the following **radius-server** commands in the Global Configuration mode:

Mode and Command	Command Purpose
awplus(config)# radius-server key <*key*>	Specifies the shared secret text string used between the switch and a RADIUS server (no default is set).
awplus(config)# radius-server retransmit <*retries*>	Specifies how many times the switch transmits each RADIUS request to the RADIUS server before giving up (the default is 3).
awplus(config)# radius-server timeout <*seconds*>	Specifies for how many seconds a switch waits for a reply to a RADIUS request before retransmitting the request.
awplus(config)# radius-server deadtime <*minutes*>	Specifies for how many minutes a RADIUS server that is not responding to authentication requests is passed over by requests for RADIUS authentication.

AAA Server Groups Configuration

Configuring the switch to use AAA server groups provides a way to group existing server hosts. This allows you to select a subset of the configured server hosts and use them for a particular service. A server group is used in conjunction with a global server-host list. The server group lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server, as long as each entry has a unique identifier. The combination of an IP address and a UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service.

To define a server host with a server group name, enter the following commands in the Global Configuration mode. The listed RADIUS server must exist in the Global Configuration mode:

Mode and Command	Command Purpose
awplus(config)# radius-server host {*<hostname>*| *<ip-address>*} [auth-port *<port-number>*] [acct-port *<port-number>*] [timeout *<seconds>*] [retransmit *<retries>*] [key *<string>*]	Specifies and defines the IP address of the server host before configuring the AAA server-group. Refer to the section **Switch to RADIUS Server Communication** of this chapter for more information on the **radius-server host** command.
awplus(config-if)# aaa group server *<group-name>*	Defines the AAA server group with a group name. This command puts the switch in server group sub configuration mode.
awplus(config-sg)# server {*<hostname>*|*<ip-address>*} [auth-port *<port-number>*] [acct-port *<port-number>*]	Associates a particular RADIUS server with the defined server group. Each security server is identified by its IP address and UDP port number. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be defined previously using the radius-server host command.

Configuring AAA server groups with deadtime

After you have configured a server host with a server name, you can use the **deadtime (RADIUS server group)** command to configure each server per server group. Configuring deadtime within a server group allows you to direct AAA traffic to separate groups of servers that have different operational characteristics.

Configuring **deadtime** is no longer limited to a global configuration. A separate timer has been attached to each server host in every server group. When a server is found to be unresponsive after numerous retransmissions and time-outs, the server is assumed to be dead. The timers attached to each server host in all server groups are triggered. In essence, the timers are checked and subsequent requests to a server, once it is assumed to be dead, are directed to alternate servers, if configured. When the network access server receives a reply from the server, it checks and stops all configured timers, if running, for that server in all server groups.

If the timer has expired, only the server to which the timer is attached is assumed to be alive. This becomes the only server that can be tried for later AAA requests using the server groups to which the timer belongs.

Note Since one server has different timers and may have different deadtime values configured in the server groups, the same server may in the future have different states, dead and alive, at the same time. To change the state of a server, you must start and stop all configured timers in all server groups.

The size of the server group will be increased because of the addition of new timers and the deadtime attribute. The overall impact of the structure depends on the number and size of the server groups and how the servers are shared among server groups in a specific configuration.

To configure deadtime within a server group, use the following commands beginning in the Global Configuration mode:

Mode and Command	Command Purpose
`awplus(config)# aaa group server radius group1`	Defines a RADIUS type server group.
`awplus(config-sg)# deadtime 1`	Configures and defines a deadtime value in minutes.
`awplus(config-sg)# exit`	Exits server group configuration mode.

Specifying RADIUS authentication

After you have identified the RADIUS server and defined the RADIUS authentication key, you must define method lists for RADIUS authentication. Because RADIUS authentication is facilitated through AAA, you must enter the **aaa authentication login** command, specifying RADIUS as the authentication method. For detailed **aaa authentication login** command information, refer to the **AAA Commands** chapter.

Specifying RADIUS accounting

The AAA accounting feature enables you to track the services users are accessing as well as the amount of network resources they are consuming. Because RADIUS accounting is facilitated through AAA, you must issue the **aaa accounting login** command, specifying RADIUS as the accounting method. For detailed **aaa accounting login** command information, refer to the **AAA Commands** chapter.

Monitoring and maintaining RADIUS

To monitor and maintain RADIUS, use the following commands in Privileged Exec mode:

Mode and Command	Command Purpose
`awplus#` `debug radius`	Displays information associated with RADIUS. For detailed **debug radius** command information, refer to the **RADIUS Commands** chapter.
`awplus#` `show radius statistics`	Displays the RADIUS statistics for accounting and authentication packets. For detailed **show radius statistics** command information, refer to the **RADIUS Commands** chapter.

RADIUS Configuration Examples

The following sections provide RADIUS configuration examples:

- RADIUS Authentication
- Single RADIUS Server Configuration
- Multiple RADIUS Server Configuration
- RADIUS Server Group Configuration
- RADIUS Server Configuration using Server Groups

RADIUS Authentication

Example The following example shows how to configure the switch to authenticate using RADIUS.

Figure 70-6: Sample RADIUS authentication to configure the switch to authenticate users

```
!
radius-server host 172.10.10.1
radius-server key radiuspass
username newuser password newpass
aaa authentication login admin
!
```

The lines in this example RADIUS authentication and accounting configuration are defined as follows:

- The **radius-server host** command defines the IP address of the RADIUS server host.
- The **radius-server key** command defines the shared secret text string between the network access server and the RADIUS server host.
- The **aaa authentication login** command defines a method list named **admin** for login authentication.

Example The following example shows how to configure the switch to authenticate logins using RADIUS.

Figure 70-7: Sample RADIUS authentication to authenticate logins

```
!
aaa authentication login radius-login group radius
!
```

This sample RADIUS authentication configuration is defined as follows:

- The **aaa authentication login radius-login group radius** command configures the switch to use RADIUS for authentication at the login prompt.

Example The following example shows how to configure the authentication method to verify a username and password at login. In this example, if a username is entered at the username prompt, that username is used for authentication.

Figure 70-8: Sample RADIUS authentication to verify a username and password

```
!
aaa authentication login default group radius
radius-server host 172.10.10.1 auth-port 1812 acct-port 1813
!
```

The lines in this sample RADIUS authentication configuration are defined as follows:

- The **aaa authentication login default group radius** command specifies that the username and password are verified by RADIUS.
- The **radius-server host 172.10.10.1 auth-port 1812 acct-port 1813** command specifies the IP address of the RADIUS server host, the UDP destination port for authentication requests, and the UDP destination port for accounting requests.

Single RADIUS Server Configuration

Example The following example shows how to configure server-specific timeout, retransmit, and key values for the RADIUS server with IP address 172.2.2.2.

Figure 70-9: Single RADIUS server sample configuration

```
!
radius-server host 172.2.2.2 timeout 5 retransmit 5 key 10
!
```

Multiple RADIUS Server Configuration

Example The following example shows how to configure two RADIUS servers with specific timeout, retransmit, and key values. The **radius-server retransmit** command changes the global retransmission value to 4 for all RADIUS servers. The **radius-server host** command configures specific timeout, retransmission, and key values for the RADIUS server hosts with IP addresses `172.2.2.2` and `172.1.1.1`

Figure 70-10: Multiple RADIUS server sample configuration

```
!
! Enable and configure radius authentication and accounting
! services on the switch:
!
aaa authentication login default group radius
aaa accounting default start-stop group radius
!
! Change the retransmission value for all RADIUS servers:
!
radius-server retransmit 4
!
! Configure per-server specific timeout, retransmission, and
! key values. Change the default auth-port and acct-port
! values.
!
radius-server host 172.2.2.2 auth-port 1645 acct-port 1646
timeout 3 retransmit 3 key radkey
!
! Configure per-server specific timeout and key values. This
! server uses the global retransmission value.
!
radius-server host 172.1.1.1 timeout 6 key rad123
!
```

RADIUS Server Group Configuration

Example The following example shows how to create server group `group2` with three RADIUS server members, each with the same IP address but with unique authentication and accounting ports.

Figure 70-11: RADIUS server group sample configuration using the same IP address

```
!
aaa group server radius group2
  server 172.1.1.1 auth-port 1645 acct-port 1646
  server 172.1.1.1 auth-port 1812 acct-port 1813
  server 172.1.1.1 auth-port 2000 acct-port 2001
!
```

RADIUS Server Configuration using Server Groups

The following example shows how to configure the network access server to recognize two different RADIUS server groups.

One of these groups, `group1`, has two different host entries on the same RADIUS server configured for the same services. The second host entry configured acts as fail over backup to the first one. Each group is individually configured for `deadtime`; `deadtime` for `group1` is one minute, and `deadtime` for `group2` is two minutes.

Figure 70-12: Multiple RADIUS servers using server groups sample configuration

```
!
! The following command configures default RADIUS parameters:
!
aaa authentication login default group group1
!
! The following commands define the group1 RADIUS server group
! and associate servers with it and configures a deadtime of
! one minute:
!
aaa group server radius group1
  server 172.1.1.1 auth-port 1645 acct-port 1646
  server 172.2.2.2 auth-port 1812 acct-port 1813
  deadtime 1
!
! The following commands define the group2 RADIUS server group
! and associate servers with it and configures a deadtime of
! two minutes:
!
aaa group server radius group2
  server 172.2.2.2 auth-port 1812 acct-port 1813
  server 172.3.3.3 auth-port 2000 acct-port 2001
  deadtime 2
!
! The following commands configure the RADIUS attributes
! for each host entry associated with one of the defined
! server groups:
!
radius-server host 172.1.1.1 auth-port 1645 acct-port 1646
radius-server host 172.2.2.2 auth-port 1812 acct-port 1813
radius-server host 172.3.3.3 auth-port 2000 acct-port 2001
!
```

Chapter 71: RADIUS Commands

Command List 71.2
deadtime (RADIUS server group) 71.2
debug radius 71.3
ip radius source-interface 71.4
radius-server deadtime 71.5
radius-server host 71.6
radius-server key 71.10
radius-server retransmit 71.11
radius-server timeout 71.12
server (Server Group) 71.14
show debugging radius 71.16
show radius 71.17
show radius statistics 71.19
undebug radius 71.19

Command List

This chapter provides an alphabetical reference for commands used to configure the device to use RADIUS servers.

deadtime (RADIUS server group)

Use this command to configure the **deadtime** parameter for the RADIUS server group. This command overrides the global dead-time configured by the **radius-server deadtime** command on page 71.5. The configured deadtime is the time period in minutes to skip a RADIUS server for authentication or accounting requests if the server is "dead". Note that a RADIUS server is considered "dead" if there is no response from the server within a defined time period.

Use the **no** variant of this command to reset the deadtime configured for the RADIUS server group. If the global deadtime for RADIUS server is configured the value will be used for the servers in the group. The global deadtime for the RADIUS server is set to 0 minutes by default.

Syntax

```
deadtime <0-1440>
no deadtime
```

Parameter	Description
<0-1440>	Amount of time in minutes.

Default The deadtime is set to 0 minutes by default.

Mode Server Group Configuration

Usage If the RADIUS server does not respond to a request packet, the packet is retransmitted the number of times configured for the **retransmit** parameter (after waiting for a **timeout** period to expire). The server is then marked "dead", and the time is recorded. The **deadtime** parameter configures the amount of time to skip a dead server; if a server is dead, no request message is sent to the server for the **deadtime** period.

Examples To configure the deadtime for 5 minutes for the RADIUS server group "GROUP1", use the command:

```
awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# server 192.168.1.1
awplus(config-sg)# deadtime 5
```

To remove the deadtime configured for the RADIUS server group "GROUP1", use the command:

```
awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# no deadtime
```

Related Commands **aaa group server**
radius-server deadtime

debug radius

This command enables RADIUS debugging. If no option is specified, all debugging options are enabled.

Use the **no** variant of this command to disable RADIUS debugging. If no option is specified, all debugging options are disabled.

Syntax

```
debug radius [packet|event|all]
no debug radius [packet|event|all]
```

Parameter	Description
`packet`	Debugging for RADIUS packets is enabled or disabled.
`event`	Debugging for RADIUS events is enabled or disabled.
`all`	Enable or disable all debugging options.

Default RADIUS debugging is disabled by default.

Mode Privileged Exec

Examples To enable debugging for RADIUS packets, use the command:

```
awplus#  debug radius packet
```

To enable debugging for RADIUS events, use the command:

```
awplus#  debug radius event
```

To disable debugging for RADIUS packets, use the command:

```
awplus# no debug radius packet
```

To disable debugging for RADIUS events, use the command:

```
awplus# no debug radius event
```

Related Commands **show debugging radius**
undebug radius

ip radius source-interface

This command configures the source IP address of every outgoing RADIUS packet to use a specific IP address or the IP address of a specific interface. If the specified interface is down or there is no IP address on the interface, then the source IP address of outgoing RADIUS packets depends on the interface the packets leave.

Use the **no** variant of this command to remove the source interface configuration. The source IP address in outgoing RADIUS packets will be the IP address of the interface from which the packets are sent.

Syntax

```
ip radius source-interface {<interface>|<ip-address>}
no ip radius source-interface
```

Parameter	Description
<interface>	Interface name.
<ip-address>	IP address in the dotted decimal format A.B.C.D.

Default Source IP address of outgoing RADIUS packets depends on the interface the packets leave.

Mode Global Configuration

Examples To configure all outgoing RADIUS packets to use the IP address of the interface "vlan1" for the source IP address, use the following commands:

```
awplus# configure terminal
awplus(config)# ip radius source-interface vlan1
```

To configure the source IP address of all outgoing RADIUS packets to use 192.168.1.10, use the following commands:

```
awplus# configure terminal
awplus(config)# ip radius source-interface 192.168.1.10
```

To reset the source interface configuration for all outgoing RADIUS packets, use the following commands:

```
awplus# configure terminal
awplus(config)# no ip radius source-interface
```

Related Commands **radius-server host**
show radius statistics

radius-server deadtime

Use this command to specify the global **deadtime** for all RADIUS servers. If a RADIUS server is considered dead, it is skipped for the specified deadtime. This command specifies for how many minutes a RADIUS server that is not responding to authentication requests is passed over by requests for RADIUS authentication.

Use the **no** variant of this command to reset the global deadtime to the default of 0 seconds, so that RADIUS servers are not skipped even if they are dead.

Syntax

```
radius-server deadtime <minutes>
no radius-server deadtime
```

Parameter	Description
<minutes>	RADIUS server deadtime in minutes in the range 0 to 1440 (24 hours).

Default The default RADIUS deadtime configured on the system is 0 seconds.

Mode Global Configuration

Usage The RADIUS client considers a RADIUS server to be dead if it fails to respond to a request after it has been retransmitted as often as specified globally by the **radius-server retransmit** command or for the server by the **radius-server host** command. To improve RADIUS response times when some servers may be unavailable, set a **deadtime** to skip dead servers.

Examples To set the dead time of the RADIUS server to 60 minutes, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server deadtime 60
```

To disable the dead time of the RADIUS server, use the following commands:

```
awplus# configure terminal
awplus(config)# no radius-server deadtime
```

Related Commands

deadtime (RADIUS server group)
radius-server host
radius-server retransmit
show radius statistics

radius-server host

Use this command to specify a remote RADIUS server host for authentication or accounting, and to set server-specific parameters. The parameters specified with this command override the corresponding global parameters for RADIUS servers. This command specifies the IP address or host name of the remote RADIUS server host and assigns authentication and accounting destination UDP port numbers.

This command adds the RADIUS server address and sets parameters to the RADIUS server. The RADIUS server is added to the running configuration after you issue this command. If parameters are not set using this command then common system settings are applied.

Use the **no** variant of this command to remove the specified server host as a RADIUS authentication and/or accounting server and set the destination port to the default RADIUS server port number (1812).

Syntax

```
radius-server host {<host-name>|<ip-address>} [acct-port <0-65535>]
   [auth-port <0-65535>] [key <key-string>] [retransmit <0-100>]
   [timeout <1-1000>]

no radius-server host {<host-name>|<ip-address>}
   [acct-port <0-65535>] [auth-port <0-65535>]
```

Parameter	Description
<host-name>	Server host name. The DNS name of the RADIUS server host.
<ip-address>	The IP address of the RADIUS server host.
acct-port	Accounting port. Specifies the UDP destination port for RADIUS accounting requests. If 0 is specified, the server is not used for accounting. The default UDP port for accounting is 1813.
<0-65535>	UDP port number (Accounting port number is set to 1813 by default) Specifies the UDP destination port for RADIUS accounting requests. If 0 is specified, the host is not used for accounting.
auth-port	Authentication port. Specifies the UDP destination port for RADIUS authentication requests. If 0 is specified, the server is not used for authentication. The default UDP port for authentication is 1812.
<0-65535>	UDP port number (Authentication port number is set to 1812 by default) Specifies the UDP destination port for RADIUS authentication requests. If 0 is specified, the host is not used for authentication.

Parameter(cont.)	Description(cont.)
`timeout`	Specifies the amount of time to wait for a response from the server. If this parameter is not specified the global value configured by the **radius-server timeout** command is used.
<1-1000>	Time in seconds to wait for a server reply (timeout is set to 5 seconds by default) The time interval (in seconds) to wait for the RADIUS server to reply before retransmitting a request or considering the server dead. This setting overrides the global value set by the **radius-server timeout** command. If no timeout value is specified for this server, the global value is used.
`retransmit`	Specifies the number of retries before skip to the next server. If this parameter is not specified the global value configured by the **radius-server retransmit** command is used.
<0-100>	Maximum number of retries (maximum number of retries is set to 3 by default) The maximum number of times to resend a RADIUS request to the server, if it does not respond within the timeout interval, before considering it dead and skipping to the next RADIUS server. This setting overrides the global setting of the **radius-server retransmit** command. If no retransmit value is specified, the global value is used.
`key`	Set shared secret key with RADIUS servers
<key-string>	Shared key string applied Specifies the shared secret authentication or encryption key for all RADIUS communications between this device and the RADIUS server. This key must match the encryption used on the RADIUS daemon. All leading spaces are ignored, but spaces within and at the end of the string are used. If spaces are used in the string, do not enclose the string in quotation marks unless the quotation marks themselves are part of the key. This setting overrides the global setting of the **radius-server key c**ommand. If no key value is specified, the global value is used.

Default The RADIUS client address is not configured (null) by default. No RADIUS server is configured.

Mode Global Configuration

Usage Multiple **radius-server host** commands can be used to specify multiple hosts. The software searches for hosts in the order they are specified. If no host-specific timeout, retransmit, or key values are specified, the global values apply to that host. If there are multiple RADIUS servers for this client, use this command multiple times—once to specify each server.

If you specify a host without specifying the auth port or the acct port, it will by default be configured for both authentication and accounting, using the default UDP ports. To set a host to be a RADIUS server for authentication requests only, set the **acct-port** parameter to 0; to set the host to be a RADIUS server for accounting requests only, set the auth-port parameter to 0.

A RADIUS server is identified by IP address, authentication port and accounting port. A single host can be configured multiple times with different authentication or accounting ports. All the RADIUS servers configured with this command are included in the predefined RADIUS server group radius, which may be used by AAA authentication, authorization and accounting commands. The client transmits (and retransmits, according to the **retransmit** and **timeout** parameters) RADIUS authentication or accounting requests to the servers in the order you specify them, until it gets a response.

Examples To add the RADIUS server `10.0.0.20`, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server host 10.0.0.20
```

To set the secret key to **allied** on the RADIUS server `10.0.0.20`, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server host 10.0.0.20 key allied
```

To delete the RADIUS server `10.0.0.20`, use the following commands:

```
awplus# configure terminal
awplus(config)# no radius-server host 10.0.0.20
```

To configure `rad1.company.com` for authentication only, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server host rad1.company.com
                acct-port 0
```

To remove the RADIUS server `rad1.company.com` configured for authentication only, use the following commands:

```
awplus# configure terminal
awplus(config)# no radius-server host rad1.company.com
                acct-port 0
```

To configure `rad2.company.com` for accounting only, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server host rad2.company.com
                auth-port 0
```

To configure 192.168.1.1 with authentication port 1000, accounting port 1001 and retransmit count 5, use the following commands:

```
awplus# configure terminal

awplus(config)# radius-server host 192.168.1.1 auth-port 1000
acct-port 1001 retransmit 5
```

Related Commands

aaa group server
radius-server key
radius-server retransmit
radius-server timeout
show radius statistics

radius-server key

This command sets a global secret key for RADIUS authentication on the switch. The shared secret text string is used for RADIUS authentication between the switch and a RADIUS server.

Note that if no secret key is explicitly specified for a RADIUS server, the global secret key will be used for the shared secret for the server.

Use the **no** variant of this command to reset the secret key to the default (null).

Syntax

```
radius-server key <key>
no radius-server key
```

Parameter	Description
<key>	Shared secret among radius server and 802.1X client.

Default The RADIUS server secret key on the system is not set by default (null).

Mode Global Configuration

Usage Use this command to set the global secret key shared between this client and its RADIUS servers. If no secret key is specified for a particular RADIUS server using the **radius-server host** command, this global key is used.

After enabling AAA authentication with the **aaa authentication login** command, set the authentication and encryption key using the **radius-server key** command so the key entered matches the key used on the RADIUS server.

Examples To set the global secret key to **allied** for RADIUS server, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server key allied
```

To set the global secret key to **secret** for RADIUS server, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server key secret
```

To delete the global secret key for RADIUS server, use the following commands:

```
awplus# configure terminal
awplus(config)# no radius-server key
```

Related Commands **radius-server host**
show radius statistics

radius-server retransmit

This command sets the retransmit counter to use RADIUS authentication on the switch. This command specifies how many times the switch transmits each RADIUS request to the RADIUS server before giving up.

This command configures the **retransmit** parameter for RADIUS servers globally. If the **retransmit** parameter is not specified for a RADIUS server by the **radius-server host** command then the global configuration set by this command is used for the server instead.

Use the **no** variant of this command to reset the re-transmit counter to the default (3).

Syntax

```
radius-server retransmit <retries>
no radius-server retransmit
```

Parameter	Description
<retries>	RADIUS server retries in the range <0-100>. The number of times a request is resent to a RADIUS server that does not respond, before the server is considered dead and the next server is tried. If no retransmit value is specified for a particular RADIUS server using the **radius-server host** command, this global value is used.

Default The default RADIUS retransmit count on the switch is 3.

Mode Global Configuration

Examples To set the RADIUS **retransmit** count to 1, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server retransmit 1
```

To set the RADIUS **retransmit** count to the default (3), use the following commands:

```
awplus# configure terminal
awplus(config)# no radius-server retransmit
```

To configure the RADIUS **retransmit** count globally with 5, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server retransmit 5
```

To disable retransmission of requests to a RADIUS server, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server retransmit 0
```

Related Commands

radius-server deadtime
radius-server host
show radius statistics

radius-server timeout

Use this command to specify the RADIUS global timeout value. This is how long the device waits for a reply to a RADIUS request before retransmitting the request, or considering the server to be dead. If no timeout is specified for the particular RADIUS server by the **radius-server host** command, it uses this global timeout value.

Note that this command configures the **timeout** parameter for RADIUS servers globally.

The **no** variant of this command resets the transmit timeout to the default (5 seconds).

Syntax

```
radius-server timeout <seconds>
no radius-server timeout
```

Parameter	Description
<seconds>	RADIUS server timeout in seconds in the range 1 to 1000. The global time in seconds to wait for a RADIUS server to reply to a request before retransmitting the request, or considering the server to be dead (depending on the **radius-server retransmit** command).

Default The default RADIUS transmit timeout on the system is 5 seconds.

Mode Global Configuration

Examples To globally set the device to wait 20 seconds before retransmitting a RADIUS request to unresponsive RADIUS servers, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server timeout 20
```

To set the RADIUS **timeout** parameter to 1 second, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server timeout 1
```

To set the RADIUS **timeout** parameter to the default (5 seconds), use the following commands:

```
awplus# configure terminal
awplus(config)# no radius-server timeout
```

To configure the RADIUS server **timeout** period globally with 3 seconds, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server timeout 3
```

To reset the global **timeout** period for RADIUS servers to the default, use the following command:

```
awplus# configure terminal
awplus(config)# no radius-server timeout
```

Related Commands

radius-server deadtime
radius-server host
radius-server retransmit
show radius statistics

server (Server Group)

This command adds a RADIUS server to a server group in Server-Group Configuration mode. The RADIUS server should be configured by the **radius-server host** command.

The server is appended to the server list of the group and the order of configuration determines the precedence of servers. If the server exists in the server group already, it will be removed before added as a new server.

The server is identified by IP address and authentication and accounting UDP port numbers. So a RADIUS server can have multiple entries in a group with different authentication and/or accounting UDP ports. The **auth-port** specifies the UDP destination port for authentication requests to the server. To disable authentication for the server, set `auth-port` to 0. If the authentication port is missing, the default port number is 1812. The **acct-port** specifies the UDP destination port for accounting requests to the server. To disable accounting for the server, set `acct-port` to 0. If the accounting port is missing, the default port number is 1812.

Use the **no** variant of this command to remove a RADIUS server from the server group.

Syntax

```
server {<hostname>|<ip-address>}
    [auth-port <0-65535>][acct-port <0-65535>]

no server {<hostname>|<ip-address>}
    [auth-port <0-65535>][acct-port <0-65535>]
```

Parameter	Description
<hostname>	Server host name
<ip-address>	Server IP address The server is identified by IP address, authentication and accounting UDP port numbers. So a RADIUS server can have multiple entries in a group with different authentication and/or accounting UDP ports.
`auth-port`	Authentication port The **auth-port** specifies the UDP destination port for authentication requests to the server. To disable authentication for the server, set **auth-port** to 0. If the authentication port is missing, the default port number is 1812.
<0-65535>	UDP port number (default: 1812)
`acct-port`	Accounting port The **acct-port** specifies the UDP destination port for accounting requests to the server. To disable accounting for the server, set **acct-port** to 0. If the accounting port is missing, the default port number is 1813.
<0-65535>	UDP port number (default: 1813)

Default The default Authentication port number is 1812 and the default Accounting port number is 1813.

Mode Server Group Configuration

Usage The RADIUS server to be added must be configured by the **radius-server host** command. In order to add or remove a server, the **auth-port** and **acct-port** parameters in this command must be the same as the corresponding parameters in the **radius-server host** command.

Examples To create a RADIUS server group RAD_AUTH1 for authentication, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa group server radius RAD_AUTH1
awplus(config-sg)# server 192.168.1.1 acct-port 0
awplus(config-sg)# server 192.168.2.1 auth-port 1000
acct-port 0
```

To create a RADIUS server group RAD_ACCT1 for accounting, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa group server radius RAD_ACCT1
awplus(config-sg)# server 192.168.2.1 auth-port 0
acct-port 1001
awplus(config-sg)# server 192.168.3.1 auth-port 0
```

To remove server 192.168.3.1 from the existing server group **GROUP1**, use the following commands:

```
awplus# configure terminal
awplus(config)# aaa group server radius GROUP1
awplus(config-sg)# no server 192.168.3.1
```

Related Commands
aaa accounting auth-mac default
aaa accounting auth-web default
aaa accounting dot1x
aaa accounting login
aaa authentication auth-mac
aaa authentication auth-web
aaa authentication login
aaa group server
radius-server host

show debugging radius

This command displays the current debugging status for the RADIUS servers.

Syntax `show debugging radius`

Mode User Exec and Privileged Exec

Example To display the current debugging status of RADIUS servers, use the command:

```
awplus# show debugging radius
```

Output **Figure 71-1: Example output from the show debugging radius command**

```
RADIUS debugging status:
RADIUS event debugging is off
RADIUS packet debugging is off
```

show radius

This command displays the current RADIUS server configuration and status.

Syntax `show radius`

Mode User Exec and Privileged Exec

Example To display the current status of RADIUS servers, use the command:

```
awplus# show radius
```

Output **Figure 71-2: Example output from the show radius command showing RADIUS servers**

```
RADIUS Global Configuration
Source Interface : not configured
Secret Key : secret
Timeout : 5 sec
Retransmit Count : 3
Deadtime : 20 min
Server Host : 192.168.1.10
Authentication Port : 1812
Accounting Port : 1813
Secret Key : secret
Timeout : 3 sec
Retransmit Count : 2
Server Host : 192.168.1.11
Authentication Port : 1812
Accounting Port : not configured
Server Name/Auth Acct Auth Acct
IP Address Port Port Status Status
-------------------------------------------------------------
192.168.1.10 1812 1813 Alive Alive
192.168.1.11 1812 N/A Alive N/A
```

Example See the sample output below showing RADIUS client status and RADIUS configuration:

```
awplus# show radius
```

Output **Figure 71-3: Example output from the show radius command showing RADIUS client status**

```
RADIUS global interface name: awplus
Secret key:
Timeout: 5
Retransmit count: 3
Deadtime: 0

Server Address: 150.87.18.89
Auth destination port: 1812
Accounting port: 1813
Secret key: swg
Timeout: 5
Retransmit count: 3
Deadtime: 0show radius local-server group
```

Output Parameter	Meaning	
`Source Interface`	The interface name or IP address to be used for the source address of all outgoing RADIUS packets.	
`Secret Key`	A shared secret key to a radius server.	
`Timeout`	A time interval in seconds.	
`Retransmit Count`	The number of retry count if a RADIUS server does not response.	
`Deadtime`	A time interval in minutes to mark a RADIUS server as “dead”.	
`Interim-Update`	A time interval in minutes to send Interim-Update Accounting report.	
`Group Deadtime`	The deadtime configured for RADIUS servers within a server group.	
`Server Host`	The RADIUS server hostname or IP address.	
`Authentication Port`	The destination UDP port for RADIUS authentication requests.	
`Accounting Port`	The destination UDP port for RADIUS accounting requests.	
`Auth Status`	The status of the authentication port. The status (“dead”, “error”, or “alive”) of the RADIUS authentication server and, if dead, how long it has been dead for.	
	`Alive`	The server is alive.
	`Error`	The server is not responding.
	`Dead`	The server is detected as dead and it will not be used for deadtime period. The time displayed in the output shows the server is in dead status for that amount of time.
	`Unknown`	The server is never used or the status is unknown.
`Acct Status`	The status of the accounting port. The status (“dead”, “error”, or “alive”) of the RADIUS accounting server and, if dead, how long it has been dead for.	

show radius statistics

This command shows the RADIUS client statistics for the switch.

Syntax `show radius statistics`

Mode User Exec and Privileged Exec

Example See the sample output below showing RADIUS client statistics and RADIUS configuration:

```
awplus# show radius statistics
```

Output **Figure 71-4: Example output from the show radius statistics command:**

```
RADIUS statistics for Server: 150.87.18.89
Access-Request Tx : 5 - Retransmit : 0
Access-Accept Rx : 1 - Access-Reject Rx : 2
Access-Challenge Rx : 2
Unknown Type : 0 - Bad Authenticator: 0
Malformed Access-Resp: 0 - Wrong Identifier: 0
Bad Attribute : 0 - Packet Dropped : 0
TimeOut : 0 - Dead count : 0
Pending Request: 0
```

undebug radius

This command applies the functionality of the **no debug radius** command on page 71.3.

Chapter 72: Local RADIUS Server Introduction and Configuration

Local RADIUS Server Introduction 72.2
Enable the Local RADIUS Server 72.2
Add the Local RADIUS Server as a RADIUS Server 72.3
Add Authenticators to the List of Authenticators 72.3
Configure the Local RADIUS Server User Database 72.4
Authenticating Login Sessions 72.5
RADIUS Authentication with User Privileges 72.5
Creating Certificates for Single Users and All Users 72.8
Defined RADIUS Attributes List 72.9

Local RADIUS Server Introduction

Local RADIUS Server provides a user authentication service feature. This feature must be enabled on the switch, because it is disabled by default. For details of commands used to configure the local RADIUS server, see **Chapter 73, Local RADIUS Server Commands**.

Enable the Local RADIUS Server

The Local RADIUS Server is disabled by default. Enter the following commands to enable the Local RADIUS Server:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# server enable
```

This will automatically initialize the internal Certificate Authority (CA) in the switch. It will also automatically create a server certificate and enroll the certificate with the Local CA by implicitly executing the following commands:

```
awplus(config)# crypto pki trustpoint local
awplus(config)# crypto pki enroll local
```

The **crypto pki trustpoint local** command declares the Local CA as the CA from which to obtain Certificates. The Local CA has be defined first so Certificates can be obtained from it. The crypto pki enroll local command obtains the system certificate from the Local CA.

The switch is automatically added to the list of authenticators that may send authentication requests to the Local RADIUS Server by implicitly executing the following commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# nas 127.0.0.1 key awplus-local-radius-
server
```

Note

The key **awplus-local-radius-server** is a pre-defined component that can be used for internal exchanges between the switch's RADIUS client and its RADIUS server.

Add the Local RADIUS Server as a RADIUS Server

Although the switch is automatically defined as a NAS (Network Access Server) for the Local RADIUS Server, you must manually add the Local RADIUS Server to the server list defined for the Local RADIUS Client.

Use the following commands to add the Local RADIUS Server as a RADIUS Server. The Local RADIUS Client can then send authentication requests to its Local RADIUS Server:

```
awplus# configure terminal
awplus(config)# radius-server host 127.0.0.1 key awplus-local-
radius-server
```

Add Authenticators to the List of Authenticators

Authenticators can send authentication requests to the Local RADIUS Server.

Use the following commands to add other authenticators to the list of authenticators.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# nas <nas-ip-address> key <nas-keystring>
```

Configure the Local RADIUS Server User Database

Add users to the RADIUS user list without assigning VLANs

For entries that will be used to authenticate dot1x supplicants, but not assign them to a VLAN, the following commands will add users to the RADIUS user list:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# user <radius-user-name> password <user-password>
```

Add users to the RADIUS user list and assign VLANs

Add users to the RADIUS user list, and define a VLAN ID that will be assigned to them.

To add entries to be used to authenticate dot1x supplicants, and assign them to a VLAN, follow the two steps shown below:

Step 1: Create groups associated with the VIDs that will be allocated

Enter the following commands to create groups with the VIDs that will be allocated to them:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group VLAN10Users
awplus(config-radsrv)# vlan 10
awplus(config-radsrv)# group VLAN11Users
awplus(config-radsrv)# vlan 11
```

Step 2: Add the users after creating groups

Add the users and refer to the relevant group in the command that creates the user as below:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# user VCSPCVLAN10 password VCSPCPass
group VLAN10Users
awplus(config-radsrv)# user VCSPCVLAN11 password VCSPCPass
group VLAN11Users
```

Authenticating Login Sessions

Authentication can be performed in multiple contexts, such as the authentication of users logging in at a console, as well as tri-authentication of devices connecting to switch ports (see **Tri-Authentication Configuration** in **Chapter 64, Authentication Introduction and Configuration**).

RADIUS Authentication with User Privileges

There are three groups of privilege levels:

- Users with privilege levels 1 to 6 have access to privilege 1 level commands.
- Users with privilege 7 to 14 have access to privilege level 1 commands and all show commands.
- Users with privilege level 15 have access to all commands.

When a user logs into a management session on a switch by console, telnet, or SSH and is being authenticated by RADIUS, the RADIUS server needs to be able to indicate to the switch what privilege level to assign to the user's session.

The way that the privilege level is associated with a user is to use the RADIUS attributes. The attributes are configured on RADIUS groups.

Because there are three group of security privilege levels, there will need to be up to three different groups for login users. Each group specifies a different privilege level.

The attributes that need to be configured on the three different RADIUS groups are as follows:

1. For the users with a privilege level of 1-6 use just the RADIUS `attribute Service-Type`, and assign it the value `NAS-Prompt-User`:

```
attribute Service-Type NAS-Prompt-User
```

2. For users with the security privilege of 7-14 use the following 2 RADIUS attributes:

```
attribute Cisco-AVPair shell:priv-lvl=7
attribute Service-Type NAS-Prompt-User
```

3. User with the administrator security privilege use just the RADIUS `attribute Service-Type`, and assign it the value `Administrative-User`:

```
attribute Service-Type Administrative-User
```

Since there is not an explicit RADIUS attribute for the users with the security privilege level 7, use "`Cisco-AVPair`" to specify this user privilege. Also, it is very important that you specify the attribute `Service-Type NAS-Prompt-User` as well, otherwise the following error is generated when a user allocated to this group tries to login into the AlliedWare Plus switch:

```
19:09:14 awplus login[16974]: Invalid user name "tests" in
main:698. Abort.
```

The RADIUS Server attribute `NAS-Prompt-User` is used for non-privileged level users as per the RADIUS RFC. This attribute is used for users with security privilege levels of 1 to 6.

Configuring these RADIUS Server attributes is achieved using Local RADIUS Server commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group users
awplus(config-radsrv-group)# attribute Service-Type NAS-
Prompt_User
```

See the below sample configuration for an AlliedWare Plus switch acting as the RADIUS Server, with the three different security privileges for `admin`, `middle-management`, and `users` groups:

Figure 72-1: Sample RADIUS Server configuration for three different security privileges:

```
crypto pki trustpoint local
!
crypto pki enroll local
radius-server local
 server enable
 nas 10.1.1.1 key test
 nas 127.0.0.1 key awplus-local-radius-server
 group admin
  attribute Service-Type Administrative-User
 group middle-management
  attribute Cisco-AVPair shell:priv-lvl=7
  attribute Service-Type NAS-Prompt-User
 group users
  attribute Service-Type NAS-Prompt-User
 user test encrypted password UukoSyvxY2v9iWXm8e/
JMDJd9iIc3RPyY09lGOb3pA4= group  users
 user tested encrypted password
sEDhM4iJRfJrLhhs+RgjpgkDXtCwuji6AllpApi9EjA= group admin
 user tests encrypted password il9aIh8JLOT6kHDV+Ix7/
8fzyfVpAwRErJg6NPQdJy8= group middle-management
```

Removing users from the RADIUS users list

To remove the user Tom from the user database of the Local RADIUS server, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no user Tom
```

Creating Certificates for Single Users and All Users

Create a certificate for a single user

A certificate for user Tom can be created from the local CA by using the commands:

```
awplus# configure terminal
awplus(config)# crypto pki enroll local user Tom
```

Create a certificate for all users

Certificates can be created for all currently defined users by using the commands:

```
awplus# configure terminal
awplus(config)# crypto pki enroll local local-radius-all-users
```

Exporting certificates

User certificates can be exported in PKCS12 format.

To export a certificate for user Tom and upload it to the TFTP server at 192.168.1.1, use the commands:

```
awplus# configure terminal
awplus(config)# crypto pki export local pkcs12 Tom tftp://
192.168.1.1/tomcert.pkcs
```

Defined RADIUS Attributes List

This is a full list of valid attributes and pre-defined values that may be used in conjunction with the **attribute** command on page 73.2, to show or configure defined RADIUS attributes.

Table 72-1 lists all Standard attributes and values, **Table 72-2** lists the Vendor-Specific attribute (attribute ID 26) names and values.

More detailed information can be found in the following RFCs, defining the attributes and values for RADIUS server:

- RFC2865: Remote Authentication Dial In User Service (RADIUS)
- RFC2866: RADIUS Accounting
- RFC2867: RADIUS Accounting Modifications for Tunnel Protocol Support
- RFC2868: RADIUS Attributes for Tunnel Protocol Support
- RFC2869: RADIUS Extensions
- RFC3162: RADIUS and IPv6
- RFC3576: Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)
- RFC3580: IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines
- RFC4072: Diameter Extensible Authentication Protocol (EAP) Application
- RFC4372: Chargeable User Identity
- RFC4603: Additional Values for the NAS-Port-Type Attribute
- RFC4675: RADIUS Attributes for Virtual LAN and Priority Support
- RFC4679: DSL Forum Vendor-Specific RADIUS Attributes
- RFC4818: RADIUS Delegated-IPv6-Prefix Attribute
- RFC4849: RADIUS Filter Rule Attribute
- RFC5176: Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)
- RFC5580: Carrying Location Objects in RADIUS and Diameter
- RFC5607: Remote Authentication Dial-In User Service (RADIUS) Authorization for Network Access Server (NAS) Management
- RFC5904: RADIUS Attributes for IEEE 802.16 Privacy Key Management Version 1 (PKMv1) Protocol Support

Table 72-1: Standard RADIUS attributes

Attribute ID and Name		Value Type/Pre-defined Values
1	User-Name	string
2	User-Password	string
3	CHAP-Password	octets (Hexadecimal string followed by 0x)
4	NAS-IP-Address	ipaddr (IPv4 address)
5	NAS-Port	Integer
6	Service-Type	Integer. Valid values are: ■ Administrative-User (6) ■ Authenticate-Only (8) ■ Authorize-Only (17) ■ Callback-Administrative (11) ■ Callback-Framed-User (4) ■ Callback-Login-User (3) ■ Callback-NAS-Prompt (9) ■ Call-Check (10) ■ Framed-Management (18) ■ Framed-User (2) ■ Login-User (1) ■ NAS-Prompt-User (7) ■ Outbound-User (5)
7	Framed-Protocol	Integer. Valid values are: ■ ARAP (3) ■ Gandalf-SLML (4) ■ PPP (1) ■ SLIP (2) ■ X.75-Synchronous (6) ■ Xylogics-IPX-SLIP (5)
8	Framed-IP-Address	ipaddr (IPv4 address)
9	Framed-IP-Netmask	ipaddr (IPv4 address)
10	Framed-Routing	integer. Valid values are: ■ Broadcast (1) ■ Broadcast-Listen (3) ■ Listen (2) ■ None (0)
11	Filter-Id	string
12	Framed-MTU	Integer
13	Framed-Compression	Integer. Valid values are: ■ IPX-Header-Compression (2) ■ None (0) ■ Stac-LZS (3) ■ Van-Jacobson-TCP-IP (1)
14	Login-IP-Host	IP Address

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
15	Login-Service	Integer. Valid values are: ■ LAT (4) ■ PortMaster (3) ■ Rlogin (1) ■ TCP-Clear (2) ■ TCP-Clear-Quiet (8) ■ Telnet (0) ■ X25-PAD (5) ■ X25-T3POS (6)
16	Login-TCP-Port	Integer. Valid values are: ■ Rlogin (513) ■ Rsh (514) ■ Telnet (23)
18	Reply-Message	string
19	Callback-Number	string
20	Callback-Id	string
22	Framed-Route	string
23	Framed-IPX-Network	IP address
24	State	octets (Hexadecimal string followed by 0x)
25	Class	octets (Hexadecimal string followed by 0x)
26	Vendor-Specific	Use the Vendor-specific Attribute Name. For valid values, see **"Vendor-specific RADIUS attributes" on page 72.19**.
27	Session-Timeout	Integer
28	Idle-Timeout	Integer
29	Termination-Action	Integer. Valid values are: ■ Default (0) ■ RADIUS-Request (1)
30	Called-Station-Id	string
31	Calling-Station-Id	string
32	NAS-Identifier	string
33	Proxy-State	octets (Hexadecimal string followed by 0x)
34	Login-LAT-Service	string
35	Login-LAT-Node	string
36	Login-LAT-Group	octets (Hexadecimal string followed by 0x)

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
37	Framed-AppleTalk-Link	Integer
38	Framed-AppleTalk-Network	Integer
39	Framed-AppleTalk-Zone	string
40	Acct-Status-Type	Integer. Valid values are: ■ Accounting-Off (8) ■ Accounting-On (7) ■ Alive (3) ■ Failed (15) ■ Interim-Update (3) ■ Start (1) ■ Stop (2) ■ Tunnel-Link-Reject (14) ■ Tunnel-Link-Start (12) ■ Tunnel-Link-Stop (13) ■ Tunnel-Reject (11) ■ Tunnel-Start (9) ■ Tunnel-Stop (10)
41	Acct-Delay-Time	Integer
42	Acct-Input-Octets	Integer
43	Acct-Output-Octets	Integer
44	Acct-Session-Id	string
45	Acct-Authentic	Integer. Valid values are: ■ Diameter (4) ■ Local (2) ■ RADIUS (1) ■ Remote (3)
46	Acct-Session-Time	Integer
47	Acct-Input-Packets	Integer
48	Acct-Output-Packets	Integer

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
49	Acct-Terminate-Cause	Integer. Valid values are: ■ Admin-Reboot (7) ■ Admin-Reset (6) ■ Callback (16) ■ Host-Request (18) ■ Idle-Timeout (4) ■ Lost-Carrier (2) ■ Lost-Service (3) ■ NAS-Error (9) ■ NAS-Reboot (11) ■ NAS-Request (10) ■ Port-Disabled (22) ■ Port-Error (8) ■ Port-Preempted (13) ■ Port-Reinit (21) ■ Port-Suspended (14) ■ Port-Unneeded (12) ■ Reauthentication-Failure (20) ■ Service-Unavailable (15) ■ Session-Timeout (5) ■ Supplicant-Restart (19) ■ User-Error (17) ■ User-Request (1)
50	Acct-Multi-Session-Id	string
51	Acct-Link-Count	Integer
52	Acct-Input-Gigawords	Integer
53	Acct-Output-Gigawords	Integer
55	Event-Timestamp	date (Not supported)
56	Egress-VLANID	Integer
57	Ingress-Filters	Integer. Valid values are: ■ Disabled (2) ■ Enabled (1)
58	Egress-VLAN-Name	string
59	User-Priority-Table	octets (Hexadecimal string followed by 0x)
60	CHAP-Challenge	octets (Hexadecimal string followed by 0x)

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
61	NAS-Port-Type	Integer. Valid values are: ■ ADSL-CAP (12) ■ ADSL-DMT (13) ■ Async (0) ■ Cable (17) ■ Ethernet (15) ■ FDDI (21) ■ G.3-Fax (10) ■ HDLC-Clear-Channel (7) ■ IDSL (14) ■ ISDN (2) ■ ISDN-V110 (4) ■ ISDN-V120 (3) ■ PIAFS (6) ■ PPPoA (30) ■ PPPoEoA (31) ■ PPPoEoE (32) ■ PPPoEoQinQ (34) ■ PPPoEoVLAN (33) ■ SDSL (11) ■ Sync (1) ■ Token-Ring (20) ■ Virtual (5) ■ Wireless-802.11 (19) ■ Wireless-Other (18) ■ X.25 (8) ■ X.75 (9) ■ xDSL (16)
62	Port-Limit	Integer
63	Login-LAT-Port	string
64	Tunnel-Type	Integer. Valid values are: ■ AH (6) ■ ATMP (4) ■ DVS (11) ■ ESP (9) ■ GRE (10) ■ IP (7) ■ IP-in-IP (12) ■ L2F (2) ■ L2TP (3) ■ MIN-IP (8) ■ PPTP (1) ■ VLAN (13) ■ VTP (5)

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
65	Tunnel-Medium-Type	Integer. Valid values are: ■ Appletalk (12) ■ Banyan-Vines (14) ■ BBN-1822 (5) ■ DecNet-IV (13) ■ E.163 (7) ■ E.164 (8) ■ E.164-NSAP (15) ■ F.69 (9) ■ HDLC (4) ■ IEEE-802 (6) ■ IP (1) ■ IPv4 (1) ■ IPv6 (2) ■ IPX (11) ■ NSAP (3) ■ X.121 (10)
66	Tunnel-Client-Endpoint	string
67	Tunnel-Server-Endpoint	string
68	Acct-Tunnel-Connection	string
69	Tunnel-Password	string
70	ARAP-Password	octets (Hexadecimal string followed by 0x)
71	ARAP-Features	octets (Hexadecimal string followed by 0x)
72	ARAP-Zone-Access	Integer. Valid values are: ■ Default-Zone (1) ■ Zone-Filter-Exclusive (4) ■ Zone-Filter-Inclusive (2)
73	ARAP-Security	Integer
74	ARAP-Security-Data	string
75	Password-Retry	integer
76	Prompt	integer. Valid values are: ■ Echo (1) ■ No-Echo (0)
77	Connect-Info	string
78	Configuration-Token	string
79	EAP-Message	octets (Hexadecimal string followed by 0x)
80	Message-Authenticator	octets (Hexadecimal string followed by 0x)
81	Tunnel-Private-Group-Id	string

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
82	Tunnel-Assignment-Id	string
83	Tunnel-Preference	Integer
84	ARAP-Challenge-Response	octets (Hexadecimal string followed by 0x)
85	Acct-Interim-Interval	Integer
86	Acct-Tunnel-Packets-Lost	Integer
87	NAS-Port-Id	string
88	Framed-Pool	string
89	Chargeable-User-Identity	string
90	Tunnel-Client-Auth-Id	string
91	Tunnel-Server-Auth-Id	string
92	NAS-Filter-Rule	string
95	NAS-IPv6-Address	ipv6addr (IPv6 address)
96	Framed-Interface-Id	ifid (Not supported)
97	Framed-IPv6-Prefix	ipv6prefix (Not supported)
98	Login-IPv6-Host	ipv6addr (IPv6 address)
99	Framed-IPv6-Route	string
100	Framed-IPv6-Pool	string
101	Error-Cause	Integer. Valid values are: ■ Administratively-Prohibited (501) ■ Invalid-Attribute-Value (407) ■ Invalid-EAP-Packet (202) ■ Invalid-Request (404) ■ Missing-Attribute (402) ■ Multiple-Session-Selection-Unsupported (508) ■ NAS-Identification-Mismatch (403) ■ Proxy-Processing-Error (505) ■ Proxy-Request-Not-Routable (502) ■ Request-Initiated (507) ■ Residual-Context-Removed (201) ■ Resources-Unavailable (506) ■ Session-Context-Not-Found (503) ■ Session-Context-Not-Removable (504) ■ Unsupported-Attribute (401) ■ Unsupported-Extension (406) ■ Unsupported-Service (405)
102	EAP-Key-Name	string

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
123	Delegated-IPv6-Prefix	ipv6prefix
126	Operator-Name	string
127	Location-Information	octets (Hexadecimal string followed by 0x)
128	Location-Data	octets (Hexadecimal string followed by 0x)
129	Basic-Location-Policy-Rules	octets (Hexadecimal string followed by 0x)
130	Extended-Location-Policy-Rules	octets (Hexadecimal string followed by 0x)
131	Location-Capable	Integer. Valid values are: ■ Civix-Location (1) ■ Geo-Location (2) ■ NAS-Location (8) ■ Users-Location (4)
132	Requested-Location-Info	Integer. Valid values are: ■ Civix-Location (1) ■ Future-Requests (16) ■ Geo-Location (2) ■ NAS-Location (8) ■ None (32) ■ Users-Location (4)
133	Framed-Management	Integer. Valid values are: ■ FTP (4) ■ Netconf (3) ■ RCP (7) ■ SCP (8) ■ SFTP (6) ■ SNMP (1) ■ TFTP (5)
134	Management-Transport-Protection	Integer. Valid values are: ■ Integrity-Confidentiality-Protection (3) ■ Integrity-Protection (2) ■ No-Protection (1)
135	Management-Policy-Id	string
136	Management-Privilege-Level	Integer
137	PKM-SS-Cert	octets (Hexadecimal string followed by 0x)
138	PKM-CA-Cert	octets (Hexadecimal string followed by 0x)
139	PKM-Config-Settings	octets (Hexadecimal string followed by 0x)
140	PKM-Cryptosuite-List	octets (Hexadecimal string followed by 0x)
141	PKM-SAID	short

Table 72-1: Standard RADIUS attributes (cont.)

Attribute ID and Name		Value Type/Pre-defined Values
142	PKM-SA-Descriptor	octets (Hexadecimal string followed by 0x)
143	PKM-Auth-Key	octets (Hexadecimal string followed by 0x)

Table 72-2: Vendor-specific RADIUS attributes

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Actual-Data-Rate-Downstream	integer
Actual-Data-Rate-Upstream	integer
Actual-Interleaving-Delay-Downstream	integer
Actual-Interleaving-Delay-Upstream	integer
ADSL-Agent-Circuit-Id	string
ADSL-Agent-Remote-Id	string
Attainable-Data-Rate-Downstream	integer
Attainable-Data-Rate-Upstream	integer
call-id	string
Cisco-Abort-Cause	string
Cisco-Account-Info	string
Cisco-Assign-IP-Pool	integer
Cisco-AVPair	string
Cisco-Call-Filter	integer
Cisco-Call-Type	string
Cisco-Command-Code	string
Cisco-Control-Info	string
Cisco-Data-Filter	integer
Cisco-Data-Rate	integer

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Cisco-Disconnect-Cause	integer. Valid values are: ■ CLID-Authentication-Failure - 4 ■ Control-C-Detected - 27 ■ EXEC-Program-Destroyed - 28 ■ Exit-Raw-TCP - 24 ■ Exit-Telnet-Session - 22 ■ Failed-PPP-CHAP-Auth - 43 ■ Failed-PPP-LCP-Negotiation - 41 ■ Failed-PPP-PAP-Auth-Fail - 42 ■ Failed-PPP-Remote-Auth - 44 ■ Idle-Timeout - 21 ■ Invalid-Protocol - 120 ■ Lost-Carrier - 1 ■ No-Carrier - 0 ■ No-Detected-Result-Codes - 2 ■ No-Remote-IP-Addr - 23 ■ Password-Fail - 25 ■ PPP-Closed-Event- 46 ■ PPP-Remote-Terminate - 45 ■ Raw-TCP-Disabled - 26 ■ Session-End-Callback - 02 ■ Session-Failed-Security - 01 ■ Session-Timeout - 00 ■ Timeout-PPP-LCP - 40 ■ Unknown - 2 ■ User-Ends-Session - 20
Cisco-Email-Server-Ack-Flag	string
Cisco-Email-Server-Address	string
Cisco-Fax-Account-Id-Origin	string
Cisco-Fax-Auth-Status	string
Cisco-Fax-Connect-Speed	string
Cisco-Fax-Coverpage-Flag	string
Cisco-Fax-Dsn-Address	string
Cisco-Fax-Dsn-Flag	string
Cisco-Fax-Mdn-Address	string
Cisco-Fax-Mdn-Flag	string
Cisco-Fax-Modem-Time	string
Cisco-Fax-Msg-Id	string
Cisco-Fax-Pages	string
Cisco-Fax-Process-Abort-Flag	string

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Cisco-Fax-Recipient-Count	string
Cisco-Gateway-Id	string
Cisco-Idle-Limit	integer
Cisco-IP-Direct	integer
Cisco-IP-Pool-Definition	string
Cisco-Link-Compression	integer
Cisco-Maximum-Channels	integer
Cisco-Maximum-Time	integer
Cisco-Multilink-ID	integer
Cisco-NAS-Port	string
Cisco-Num-In-Multilink	integer
Cisco-Policy-Down	string
Cisco-Policy-Up	string
Cisco-Port-Used	string
Cisco-PPP-Async-Map	integer
Cisco-PPP-VJ-Slot-Comp	integer
Cisco-Pre-Input-Octets	integer
Cisco-Pre-Input-Packets	integer
Cisco-Pre-Output-Octets	integer
Cisco-Pre-Output-Packets	integer
Cisco-PreSession-Time	integer
Cisco-PW-Lifetime	integer
Cisco-Route-IP	integer
Cisco-Service-Info	string
Cisco-Subscriber-Password	string
Cisco-Target-Util	integer
Cisco-Xmit-Rate	integer
gw-final-xlated-cdn	string

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
gw-final-xlated-cgn	string
gw-rxd-cdn	string
gw-rxd-cgn	string
h323-billing-model	string
h323-call-origin	string
h323-call-type	string
h323-conf-id	string
h323-connect-time	string
h323-credit-amount	string
h323-credit-time	string
h323-currency	string
h323-disconnect-cause	string
h323-disconnect-time	string
h323-gw-id	string
h323-incoming-conf-id	string
h323-preferred-lang	string
h323-prompt-id	string
h323-redirect-ip-address	string
h323-redirect-number	string
h323-remote-address	string
h323-return-code	string
h323-setup-time	string
h323-time-and-day	string
h323-voice-quality	string
incoming-req-uri	string
IWF-Session	octets
Maximum-Data-Rate-Downstream	integer
Maximum-Data-Rate-Upstream	integer

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
Maximum-Interleaving-Delay-Downstream	integer
Maximum-Interleaving-Delay-Upstream	integer
method	string
Minimum-Data-Rate-Downstream	integer
Minimum-Data-Rate-Downstream-Low-Power	integer
Minimum-Data-Rate-Upstream	integer
Minimum-Data-Rate-Upstream-Low-Power	integer
MS-Acct-Auth-Type	integer. Valid values are: ■ CHAP - 2 ■ EAP - 5 ■ MS-CHAP-1 - 3 ■ MS-CHAP-2 - 4 ■ PAP - 1
MS-Acct-EAP-Type	integer. Valid values are: ■ Generic-Token-Card - 6 ■ MD5 - 4 ■ OTP - 5 ■ TLS -13
MS-AFW-Protection-Level	integer. Valid values are: ■ HECP-Response-Sign-And-Encrypt - 2 ■ HECP-Response-Sign-Only - 1
MS-AFW-Zone	integer. Valid values are: ■ MS-AFW-Zone-Boundary-Policy - 1 ■ MS-AFW-Zone-Protected-Policy - 3 ■ MS-AFW-Zone-Unprotected-Policy - 2
MS-ARAP-PW-Change-Reason	integer. Valid values are: ■ Admin-Requires-Password-Change - 3 ■ Expired-Password - 2 ■ Just-Change-Password - 1 ■ Password-Too-Short - 4
MS-BAP-Usage	integer. Valid values are: ■ Allowed - 1 ■ Not-Allowed - 0 ■ Required - 2
MS-CHAP2-CPW	octets
MS-CHAP2-Response	octets
MS-CHAP2-Success	octets

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
MS-CHAP-Challenge	octets
MS-CHAP-CPW-1	octets
MS-CHAP-CPW-2	octets
MS-CHAP-Domain	string
MS-CHAP-Error	string
MS-CHAP-LM-Enc-PW	octets
MS-CHAP-MPPE-Keys	octets
MS-CHAP-NT-Enc-PW	octets
MS-CHAP-Response	octets
MS-Extended-Quarantine-State	integer. Valid values are: ■ Infected - 2 ■ No-Data - 4 ■ Transition - 1 ■ Unknown - 3
MS-Filter	octets
MS-HCAP-Location-Group-Name	string
MS-HCAP-User-Groups	string
MS-HCAP-User-Name	string
MS-Identity-Type	integer. Valid values are: ■ Ignore-User-Lookup-Failure - 2 ■ Machine-Health-Check - 1
MS-IPv4-Remediation-Servers	octets
MS-IPv6-Filter	octets
MS-IPv6-Remediation-Servers	octets
MS-Link-Drop-Time-Limit	integer
MS-Link-Utilization-Threshold	integer
MS-Machine-Name	string
MS-MPPE-Encryption-Policy	octets
MS-MPPE-Encryption-Type	octets
MS-MPPE-Encryption-Types	octets
MS-MPPE-Recv-Key	octets

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
MS-MPPE-Send-Key	octets
MS-Network-Access-Server-Type	integer. Valid values are: ■ DHCP-Server - 3 ■ HCAP-Server - 6 ■ HRA - 5 ■ Remote-Access-Server - 2 ■ Terminal-Server-Gateway - 1 ■ Unspecified - 0 ■ Wireless-Access-Point - 4
MS-New-ARAP-Password	octets
MS-Old-ARAP-Password	octets
MS-Primary-DNS-Server	ipaddr
MS-Primary-NBNS-Server	ipaddr
MS-Quarantine-Grace-Time	integer
MS-Quarantine-IPFilter	octets
MS-Quarantine-Session-Timeout	integer
MS-Quarantine-SOH	octets
MS-Quarantine-State	integer. Valid values are: ■ Full-Access - 0 ■ Probation - 2 ■ Quarantine - 1
MS-Quarantine-User-Class	string
MS-RAS-Client-Name	string
MS-RAS-Client-Version	string
MS-RAS-Correlation	octets
MS-RAS-Vendor	integer
MS-RAS-Version	string
MS-RNAP-Not-Quarantine-Capable	integer. Valid values are: ■ SoH-Not-Sent - 1 ■ SoH-Sent - 0
MS-Secondary-DNS-Server	ipaddr
MS-Secondary-NBNS-Server	ipaddr
MS-Service-Class	string
MS-TSG-Device-Redirection	integer

Table 72-2: Vendor-specific RADIUS attributes (cont.)

Vendor-Specific Attribute Name	Value Type/Pre-defined Value
MS-User-IPv4-Address	ipaddr
MS-User-IPv6-Address	ipv6addr
MS-User-Security-Identity	string
next-hop-dn	string
next-hop-ip	string
outgoing-req-uri	string
prev-hop-ip	string
prev-hop-via	string
release-source	string
remote-media-address	string
session-protocol	string
sip-conf-id	string
sip-hdr	string
subscriber	string

Chapter 73: Local RADIUS Server Commands

Command List 73.2
attribute 73.2
authentication 73.5
clear radius local-server statistics 73.6
copy fdb-radius-users (to file) 73.7
copy local-radius-user-db (from file) 73.9
copy local-radius-user-db (to file) 73.10
crypto pki enroll local 73.11
crypto pki enroll local local-radius-all-users 73.12
crypto pki enroll local user 73.13
crypto pki export local pem 73.14
crypto pki export local pkcs12 ..... 73.15
crypto pki trustpoint local 73.16
debug crypto pki 73.17
domain-style 73.18
egress-vlan-id 73.19
egress-vlan-name 73.20
group 73.21
nas 73.22
radius-server local 73.23
server auth-port 73.24
server enable 73.25
show crypto pki certificates 73.26
show crypto pki certificates local-radius-all-users 73.28
show crypto pki certificates user 73.29
show crypto pki trustpoints 73.30
show radius local-server group 73.31
show radius local-server nas 73.32
show radius local-server statistics 73.33
show radius local-server user 73.34
user (RADIUS server) 73.36
vlan (RADIUS server) 73.38

Command List

This chapter provides an alphabetical reference for commands used to configure the local RADIUS server on the device. For more information, see **Chapter 72, Local RADIUS Server Introduction and Configuration**.

attribute

Use this command to define a RADIUS attribute for the local RADIUS server user group.

For a complete list of defined RADIUS attributes and values, see **"Defined RADIUS Attributes List" on page 72.9**.

When used with the **help** parameter the **attribute** command displays a list of standard and vendor specific valid RADIUS attributes that are supported by the local RADIUS server.

If an attribute name is specified with the **help** parameter, then the **attribute** command displays a list of predefined attribute names. Note that you can only use the defined RADIUS attribute names and not define your own.

When used with the **value** parameter the **attribute** command configures RADIUS attributes to the user group. If the specified attribute is already defined then it is replaced with the new value.

Use the **no** variant of this command to delete an attribute from the local RADIUS server user group.

Syntax

```
attribute [<attribute-name>|<attribute-id>] help
attribute {<attribute-name>|<attribute-id>} <value>
no attribute {<attribute-name>|<attribute-id>}
```

Parameter	Description
<attribute-name>	RADIUS attribute name for standard attributes (see **Table 72-1 on page 72.10**) or Vendor-Specific attributes (see **Table 72-2 on page 72.19**).
<attribute-id>	RADIUS attribute numeric identifier for standard attributes (**Table 72-1 on page 72.10**).
<value>	RADIUS attribute value.
`help`	Display a list of available attribute types.

Default By default, no attributes are configured.

Mode RADIUS Server Group Configuration

Usage For the Standard attributes, the attribute may be specified using either the attribute name, or its numeric identifier. For example, command:

```
awplus(config-radsrv-group)# attribute acct-terminate-cause
                             help
```

will produce the same results as command:

```
awplus(config-radsrv-group)# attribute 49 help
```

In the same way, where the specific attribute has a pre-defined value, the parameter <*value*> may be substituted with the Value Name or with its numeric value, for example command:

```
awplus(config-radsrv-group)# attribute acct-terminate-cause
                             user-request
```

will produce the same results as command:

```
awplus(config-radsrv-group)# attribute 49 1
```

or command:

```
awplus(config-radsrv-group)# attribute acct-terminate-cause 1
```

Examples To check a list of all available defined RADIUS attribute names, use the following commands:

```
                      awplus# configure terminal
                awplus(config)# radius-server local
          awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# attribute help
```

A list of Vendor-specific Attributes displays after the list of defined Standard Attributes.

To get help for valid RADIUS attribute values for the attribute `Service-Type`, use the following commands:

```
                      awplus# configure terminal
                awplus(config)# radius-server local
          awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# attribute Service-Type help
```

```
Service-Type : integer (Integer number)
Pre-defined values :
    Administrative-User (6)
    Authenticate-Only (8)
    Authorize-Only (17)
    Callback-Administrative (11)
    Callback-Framed-User (4)
    Callback-Login-User (3)
    Callback-NAS-Prompt (9)
    Call-Check (10)
    Framed-User (2)
    Login-User (1)
    NAS-Prompt-User (7)
    Outbound-User (5)
```

To define the attribute name 'Service-Type' with Administrative User (6) to the RADIUS User Group 'Admin', use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# attribute Service-Type 6
```

To delete the attribute 'Service-Type' from the RADIUS User Group 'Admin', use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group Admin
awplus(config-radsrv-group)# no attribute Service-Type
```

Related Commands **egress-vlan-id**
egress-vlan-name

authentication

Use this command to enable the specified authentication methods on the local RADIUS server.

Use the **no** variant of this command to disable specified authentication methods on the local RADIUS server.

Syntax

```
authentication {mac|eapmd5|eaptls|peap}
no authentication {mac|eapmd5|eaptls|peap}
```

Parameter	Description
`mac`	Enable MAC authentication method.
`eapmd5`	Enable EAP-MD5 authentication method.
`eaptls`	Enable EAP-TLS authentication method.
`peap`	Enable EAP-PEAP authentication method.

Default All authentication methods are enabled by default.

Mode RADIUS Server Configuration

Examples The following commands enable EAP-MD5 authentication methods on the local RADIUS server.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# authentication eapmd5
```

The following commands disable EAP-MD5 authentication methods on Local RADIUS server.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no authentication eapmd5
```

Related Commands **server enable**
show radius local-server statistics

clear radius local-server statistics

Use this command to clear the statistics stored on the switch for the local RADIUS server.

Use this command without any parameters to clear all types of local RADIUS server statistics.

Syntax `clear radius local-server statistics [nas|server|user]`

Parameter	Description
`nas`	Clear the NAS (Network Access Server) statistics on the switch. For example, clearing statistics stored for NAS server invalid passwords.
`server`	Clear the Local RADIUS Server statistics on the switch. For example, clearing Local RADIUS Servers statistics for all failed login attempts.
`user`	Clear the Local RADIUS Server user statistics. For example, clearing statistics stored for the number of successful user logins.

Mode Privileged Exec

Usage Refer to the sample output for the **show radius local-server statistics** for further information about the type of statistics each parameter option for this command clears. Both the **nas** and **server** parameters clear unknown username and invalid passwords statistics, while the `user` parameter clears the number of successful and failed logins for each local RADIUS server user.

Examples To clear the NAS (Network Access Server) statistics stored on the switch, use the command:

```
awplus# clear radius local-server statistics nas
```

To clear the local RADIUS server statistics stored on the switch, use the command:

```
awplus# clear radius local-server statistics server
```

To clear the local RADIUS server user statistics stored on the switch, use the command:

```
awplus# clear radius local-server statistics user
```

Related Commands **show radius local-server statistics**

copy fdb-radius-users (to file)

Use this command to create a set of local RADIUS server users from MAC addresses in the local FDB. A local RADIUS server user created using this command can be used for MAC authentication.

Syntax

```
copy fdb-radius-users {local-radius-user-db|flash|nvs|usb|debug|tftp|
   scp|fserver|<url>} [interface <port>] [vlan <vid>] [group <name>]
   [export-vlan [<radius-group-name>]]
```

Parameter	Description
`local-radius-user-db`	Copy the local RADIUS server users created to the local RADIUS server.
`flash`	Copy the local RADIUS server users created to Flash memory.
`nvs`	Copy the local RADIUS server users created to NVS memory.
`usb`	Copy the local RADIUS server users created to USB storage device.
`debug`	Copy the local RADIUS server users created to debug.
`tftp`	Copy the local RADIUS server users created to the TFTP destination.
`scp`	Copy the local RADIUS server users created to the SCP destination.
`fserver`	Copy the local RADIUS server users created to the remote file server.
`<url>`	Copy the local RADIUS server users created to the specified URL.
`interface <port>`	Copy only MAC addresses learned on a specified switch port. Wildcards may be used when specifying an interface name.For example, when you specify interface port1.* then this command generates RADIUS server users for MAC addresses learned on stack 1.
`vlan <vid>`	Copy only MAC addresses learned on a specified VLAN.
`group <name>`	Assign a group name to the local RADIUS server users created.
`export-vlan`	Export VLAN ID assigned to exported FDB entry.
`<radius-group-name>`	Prefix for Radius group name storing VLAN ID

Mode Privileged Exec

Usage The local RADIUS server users created are written to a specified destination file in local RADIUS user CSV (Comma Separated Values) format. The local RADIUS server users can then be imported to a local RADIUS server using the **copy local-radius-user-db (from file)** command.

The name and password of the local RADIUS server users created use a MAC address, which can be used for MAC authentication.

This command does not copy a MAC address learned by the CPU or the management port.

This command can filter FDB entries by the interface name and the VLAN ID. When the interface name and the VLAN ID are specified, this command generates local RADIUS server users from only the MAC address learned on the specified interface and on the specified VLAN.

Examples To register the local RADIUS server users from the local FDB directly to the local RADIUS server, use the command:

```
awplus# copy fdb-radius-users local-radius-user-db
```

To register the local RADIUS server users from the interface `port1.0.1` to the local RADIUS server, use the command:

```
awplus# copy fdb-radius-users local-radius-user-db interface
port1.0.1
```

To copy output generated as local RADIUS server user data from MAC addresses learned on `vlan10` on interface `port1.0.1` to the file `radius-user.csv`, use the command:

```
awplus# copy fdb-radius-users radius-user.csv interface
port1.0.1 vlan10
```

To copy output generated as local RADIUS server user data from MAC addresses learned on `vlan10` on interface `port1.0.1` to a file on the remote file server, use the command:

```
awplus# copy fdb-radius-users fserver interface port1.0.1
vlan10
```

Related Commands **copy local-radius-user-db (to file)**
copy local-radius-user-db (from file)

copy local-radius-user-db (from file)

Use this command to copy the Local RADIUS server user data from a file. The file, including the RADIUS user data in the file, must be in the CSV (Comma Separated Values) format.

You can select **add** or **replace** as the copy method. The **add** parameter option copies the contents of specified file to the local RADIUS server user database. If the same user exists then the old user is removed before adding a new user. The **replace** parameter option deletes all contents of the local RADIUS server user database before copying the contents of specified file.

Syntax

```
copy <source-url> local-radius-user-db [add|replace]
```

Parameter	Description
<source-url>	URL of the source file.
`add`	Add file contents to local RADIUS server user database.
`replace`	Replace current local RADIUS server user database with file contents.

Default When no copy method is specified with this command the **replace** option is applied.

Mode Privileged Exec

Examples To replace the current local RADIUS server user data to the contents of http://datahost/user.csv, use the following command:

```
awplus# copy http://datahost/user.csv local-radius-user-db
```

To add the contents of http://datahost/user.csv to the current local RADIUS server user database, use the following command:

```
awplus# copy http://datahost/user.csv local-radius-user-db add
```

Related commands

copy fdb-radius-users (to file)
copy local-radius-user-db (to file)

copy local-radius-user-db (to file)

Use this command to copy the local RADIUS server user data to a file. The output file produced is CSV (Comma Separated Values) format.

Syntax `copy local-radius-user-db {flash|nvs|usb|tftp|scp|<destination-url>}`

Parameter	Description
`flash`	Copy to flash memory.
`nvs`	Copy to NVS memory.
`usb`	Copy to USB storage device.
`tftp`	Copy to TFTP destination.
`scp`	Copy to SCP destination.
<destination-url>	URL of the Destination file.

Mode Privileged Exec

Example Copy the current local RADIUS server user data to http://datahost/user.csv.

```
awplus# copy local-radius-user-db http://datahost/user.csv
```

Related Commands **copy fdb-radius-users (to file)**
copy local-radius-user-db (from file)

crypto pki enroll local

Use this command to obtain a system certificate from the Local CA (Certificate Authority).

Use the **no** variant of this command to delete system certificates created by a Local CA (Certificate Authority).

Syntax

```
crypto pki enroll local
no crypto pki enroll local
```

Default The system certificate is not available until this command is issued.

Mode Global Configuration

Examples The following command obtains the system certificate from the Local CA (Certificate Authority).

```
awplus# configure terminal
awplus(config)# crypto pki enroll local
```

The following command deletes the system certificate created by the Local CA (Certificate Authority).

```
awplus# configure terminal
awplus(config)# no crypto pki enroll local
```

Related Commands **crypto pki trustpoint local**
group

crypto pki enroll local local-radius-all-users

Use this command to create certificates for all users registered in the local RADIUS server. These certificates are created by the Local Certificate Authority (CA) on the switch.

Syntax `crypto pki enroll local local-radius-all-users`

Default By default, there are no certificates for users in the local RADIUS server.

Mode Global Configuration

Example The following command obtains the local RADIUS server certificates for the user from the Local CA (Certificate Authority).

```
awplus# configure terminal
awplus(config)# crypto pki enroll local local-radius-all-users
```

Related Commands **crypto pki trustpoint local**
show crypto pki certificates

crypto pki enroll local user

Use this command to obtain a local user certificate from the Local CA (Certificate Authority).

Use the **no** variant of this command to delete user certificates created by the Local CA (Certificate Authority).

Syntax

```
crypto pki enroll local user <user-name>
no crypto pki enroll local user <user-name>
```

Parameter	Description
<user-name>	User name.

Default By default, there is no user certificate.

Mode Global Configuration

Examples The following command obtains Tom's certificate from the Local CA (Certificate Authority).

```
awplus# configure terminal
awplus(config)# crypto pki enroll local user Tom
```

The following command deletes Tom's certificates created by the Local CA (Certificate Authority):

```
awplus# configure terminal
awplus(config)# no crypto pki enroll local user Tom
```

Related Commands **crypto pki trustpoint local**
show crypto pki certificates

crypto pki export local pem

Use this command to export the certificate associated with the Local CA to a PEM format file.

Syntax `crypto pki export local pem url <url>`

Parameter	Description
<url>	URL string.

Mode Global Configuration

Example The following command exports the Local CA certificate to a PEM format file.

```
awplus# configure terminal
awplus(config)# crypto pki export local pem url tftp://
192.168.1.1/cacert.pem
```

Related Commands crypto pki enroll local

crypto pki export local pkcs12

Use this command to export a specified certificate to a PKCS12 format file.

This command cannot be used for exporting certificates for the local system.

Syntax `crypto pki export local pkcs12 <user-name> <destination-url>`

Parameter	Description
<user-name>	User name.
<destination-url>	Destination URL string.

Mode Global Configuration

Examples The following commands exports a certificate for a user named **client** to a PKCS12 format file.

```
awplus# configure terminal
awplus(config)# crypto pki export local pkcs12 client tftp://
192.168.1.1/cacert.pem
```

To export Tom's certificate to PKSC12 format file, use the commands:

```
awplus# configure terminal
awplus(config)# crypto pki export local pksc12 Tom tftp://
192.168.1.1/tom.pfx
```

Related Commands **crypto pki enroll local**

crypto pki trustpoint local

Use this command to declare the Local CA (Certificate Authority) as the trustpoint that the system uses. The ca-trustpoint configuration mode is available after this command is issued.

Use the **no** variant of this command to delete all information and certificates associated with Local CA as the trustpoint.

Syntax

```
crypto pki trustpoint local
no crypto pki trustpoint local
```

Default Local CA is not a trustpoint.

Mode Global Configuration

Examples Use the following commands to declare the Local CA as the trustpoint.

```
awplus# configure terminal
awplus(config)# crypto pki trustpoint local
```

Use the following commands to delete all information and certificates associated with the Local CA.

```
awplus# configure terminal
awplus(config)# no crypto pki trustpoint local
```

To create a client certificate for all users registered to the local RADIUS server, use the following commands:

```
awplus(config)# crypto pki trustpoint local
awplus(ca-trust-point)# exit
awplus(config)# crypto pki enroll local alternative
```

Related Commands

crypto pki enroll local
show crypto pki trustpoints

debug crypto pki

Use this command to enable Public Key Infrastructure (PKI) debugging. When PKI debugging is enabled, the PKI module starts generating diagnostic messages to the system log.

Use the **no** variant of this command to disable Public Key Infrastructure (PKI) debugging. When PKI debugging is disabled, the PKI module stops generating diagnostic messages to the system log.

Syntax

```
debug crypto pki
no debug crypto pki
```

Default PKI debugging is disabled by default

Mode Privileged Exec

Examples To enable the PKI debugging facility, use the command:

```
awplus# debug crypto pki
```

To disable the PKI debugging facility, use the command:

```
awplus# no debug crypto pki
```

domain-style

Use this command to enable a specified domain style on the local RADIUS server. The local RADIUS server decodes the domain portion of a username login string when this command is enabled.

Use the **no** variant of this command to disable the specified domain style on the local RADIUS server.

Syntax `domain-style {suffix-atsign|ntdomain}`

Parameter	Description
`suffix-atsign`	Enable at sign "@" delimited suffix style, i.e. "user@domain".
`ntdomain`	Enable NT domain style, i.e. "domain\user".

Default This feature is disabled by default.

Mode RADIUS Server Configuration

Usage When both domain styles are enabled, the first domain style configured has the highest priority. A username login string is matched against the first domain style enabled. Then, if the username login string is not decoded, it is matched against the second domain style enabled.

Examples To enable NT domain style on the local RADIUS server, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# domain-style ntdomain
```

To disable NT domain style on the local RADIUS server, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no domain-style ntdomain
```

Related Commands **server enable**

egress-vlan-id

Use this command to configure the standard RADIUS attribute "Egress-VLANID (56)" for the local RADIUS Server user group.

Use the **no** variant of this command to remove the Egress-VLANID attribute from the local RADIUS server user group.

Syntax

```
egress-vlan-id <vid> [tagged|untagged]
no egress-vlan-id
```

Parameter	Description
<vid>	The VLAN identifier to be used for the Egress VLANID attribute, in the range 1 to 4094.
`tagged`	Set frames on the VLAN as tagged. This sets the tag indication field to indicate that all frames on this VLAN are tagged.
`untagged`	Set all frames on the VLAN as untagged. This sets the tag indication field to indicate that all frames on this VLAN are untagged.

Default By default, no Egress-VLANID attributes are configured.

Mode RADIUS Server Group Configuration

Usage When a Voice VLAN is configured for dynamic VLAN allocation (**switchport voice vlan** command on page 19.33), the RADIUS server must be configured to send the VLAN information when an IP phone is successfully authenticated. Use either the **egress-vlan-id** command or the **egress-vlan-name** command on page 73.20, and specify the **tagged** parameter.

Examples To set the "Egress-VLANID" attribute for the NormalUsers local RADIUS server user group to VLAN identifier 200, with tagged frames, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# egress-vlan-id 200 tagged
```

To remove the "Egress-VLANID" attribute for the NormalUsers local RADIUS server user group, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# no egress-vlan-id
```

Related Commands

attribute
egress-vlan-name
switchport voice vlan

egress-vlan-name

Use this command to configure the standard RADIUS attribute "Egress-VLAN-Name (58)" for the local RADIUS server user group.

Use the **no** variant of this command to remove the Egress-VLAN-Name attribute from the local RADIUS server user group.

Syntax

```
egress-vlan-name <vlan-name> [tagged|untagged]
no egress-vlan-name
```

Parameter	Description
<vlan-name>	The VLAN name to be configured as the Egress-VLAN-Name attribute.
`tagged`	Set frames on the VLAN as tagged. This sets the tag indication field to indicate that all frames on this VLAN are tagged.
`untagged`	Set all frames on the VLAN as untagged. This sets the tag indication field to indicate that all frames on this VLAN are untagged.

Default By default, no Egress-VLAN-Name attributes are configured.

Mode RADIUS Server Group Configuration

Usage When a Voice VLAN is configured for dynamic VLAN allocation (**switchport voice vlan** command on page 19.33), the RADIUS server must be configured to send the VLAN information when an IP phone is successfully authenticated. Use either the **egress-vlan-id** command on page 73.19 or the **egress-vlan-name** command, and specify the **tagged** parameter.

Examples To configure the "Egress-VLAN-Name" attribute for the RADIUS server user group NormalUsers with the VLAN name "vlan2" and all frames on this VLAN tagged, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# egress-vlan-name vlan2 tagged
```

To delete the "Egress-VLAN-Name" attribute for the NormalUsers group, use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# no egress-vlan-name
```

Related Commands
attribute
egress-vlan-id
switchport voice vlan

group

Use this command to create a local RADIUS server user group, and enter local RADIUS Server User Group Configuration mode.

Use the **no** variant of this command to delete the local RADIUS server user group.

Syntax

```
group <user-group-name>
no group <user-group-name>
```

Parameter	Description
<user-group-name>	User group name string.

Mode RADIUS Server Configuration

Examples The following command creates the user group NormalUsers.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
```

The following command deletes user group NormalUsers.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no group NormalUsers
```

Related Commands

user (RADIUS server)
show radius local-server user
vlan (RADIUS server)

nas

This command adds a client device (the Network Access Server or the NAS) to the list of devices that are able to send authentication requests to the local RADIUS server. The NAS is identified by its IP address and a shared secret (also referred to as a shared key) must be defined that the NAS will use to establish its identity.

Use the **no** variant of this command to remove a NAS client from the list of devices that are allowed to send authentication requests to the local RADIUS server.

Syntax

```
nas <ip-address> key <nas-keystring>
no nas <ip-address>
```

Parameter	Description
<ip-address>	RADIUS NAS IP address.
<nas-keystring>	NAS shared keystring.

Mode RADIUS Server Configuration

Examples The following commands add the NAS with an IP address of `192.168.1.2` to the list of clients that may send authentication requests to the local RADIUS server. Note the shared key that this NAS will use to establish its identify is `NAS_PASSWORD`.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# nas 192.168.1.2 key NAS_PASSWORD
```

The following commands remove the NAS with an IP address of `192.168.1.2` from the list of clients that are allowed to send authentication requests to the local RADIUS server:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no nas 192.168.1.2
```

Related Commands **show radius local-server nas**

radius-server local

Use this command to navigate to the Local RADIUS server configuration mode `(config-radsrv)` from the Global Configuration mode `(config)`.

Syntax `radius-server local`

Mode Global Configuration

Example Local RADIUS Server commands are available from `config-radsrv` configuration mode. To change mode from User Exec mode to the Local RADIUS Server mode (config-radsrv), use the commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)#
```

Output

```
awplus(config)#radius-server local
Creating Local CA repository.....OK
Enrolling Local System to local trustpoint..OK
awplus(config-radsrv)#
```

Related Commands

server enable
show radius local-server group
show radius local-server nas
show radius local-server statistics
show radius local-server user

server auth-port

Use this command to change the UDP port number for local RADIUS server authentication.

Use the **no** variant of this command to reset the RADIUS server authentication port back to the default.

Syntax

```
server auth-port <1-65535>
no server auth-port
```

Parameter	Description
<1-65535>	UDP port number.

Default The default local RADIUS server UDP authentication port number is 1812.

Mode RADIUS Server Configuration

Examples The following commands set the RADIUS server authentication port to 10000.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# server port 10000
```

The following commands reset the RADIUS server authentication port back to the default UDP port of 1812.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no server port
```

Related Commands **server enable**
show radius local-server statistics

server enable

This command enables the local RADIUS server. The local RADIUS server feature is started immediately when this command is issued.

The **no** variant of this command disables local RADIUS server. When this command is issued, the local RADIUS server stops operating.

Syntax

```
server enable
no server enable
```

Default The local RADIUS server is disabled by default and must be enabled for use with this command.

Mode RADIUS Server Configuration

Examples To enable the local RADIUS server, use the following commands:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# server enable
```

To disable the local RADIUS server, use the command:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no server enable
```

Related Commands

server auth-port
show radius local-server statistics

show crypto pki certificates

Use this command to display certificate information for Local CA and Local System certificates.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show crypto pki certificates [local-ca|local]`

Parameter	Description
`local-ca`	Local CA certificate.
`local`	Local system certificate.

Mode User Exec and Privileged Exec

Examples The following command displays Local CA (Certificate Authority) certificate information.

```
awplus# show crypto pki certificates local-ca
```

The following command displays Local System certificate information.

```
awplus# show crypto pki certificates local
```

The following command displays information for all Local CA and Local System certificates.

```
awplus# show crypto pki certificates
```

Output

Figure 73-1: Example output from the show crypto pki certificates command showing Local System and Local CA certificates

```
awplus#show crypto pki certificates
Certificate: Local System
    Data:
        Version: 3 (0x2)
        Serial Number: 4 (0x4)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:50:55 2009 GMT
            Not After : Oct  6 07:50:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
Certificate: Local CA
    Data:
        Version: 3 (0x2)
        Serial Number: 0 (0x0)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:55:55 2009 GMT
            Not After : Oct  6 07:55:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
```

Table 73-1: Parameters in the output of the show crypto pki certificates command

Parameter	Description
`Certificate`	Certificate name.
`Version`	Protocol version.
`Serial Number`	Serial number of the certificate.
`Signature Algorithm`	Algorithm used for the certificate signature.
`Issuer`	Subject of issuer creating the certificate.
`Validity`	Validity period.
`Subject`	Subject of the certificate.

Related Commands **crypto pki enroll local**

show crypto pki certificates local-radius-all-users

Use this command to display certificate information for local RADIUS server users.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show crypto pki certificates local-radius-all-users`

Mode User Exec and Privileged Exec

Example The following command displays information of all local RADIUS server user certificates.

```
awplus# show crypto pki certificates local-radius-all-users
```

Output

Figure 73-2: Example output from the show crypto pki certificates local-radius-all-users command

```
awplus#show crypto pki certificates local-radius-all-users
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:50:55 2009 GMT
            Not After : Oct  6 07:50:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
```

Table 73-2: Parameters in the output of the show crypto pki certificates local-radius-all-users command

Parameter	Description
`Certificate`	Certificate name.
`Version`	Protocol version.
`Serial Number`	Serial number of the certificate.
`Signature Algorithm`	Algorithm used for the certificate signature.
`Issuer`	Subject of issuer creating the certificate.
`Validity`	Validity period.
`Subject`	Subject of the certificate.

Related Commands crypto pki enroll local local-radius-all-users

show crypto pki certificates user

Use this command to display certificate information for a specified local RADIUS server user.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show crypto pki certificates user [<user-name>]`

Parameter	Description
<user-name>	User name.

Mode User Exec and Privileged Exec

Example The following command displays Tom's certificate information.

```
awplus# show crypto pki certificates user Tom
```

Output

Figure 73-3: Example output from the show crypto pki certificates user command to show certificate information for user Tom

```
awplus#show crypto pki certificates user Tom
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 2 (0x2)
        Signature Algorithm: sha1WithRSAEncryption
        Issuer: O=Allied-Telesis, CN=AlliedwarePlusCA
        Validity
            Not Before: Oct  8 07:50:55 2009 GMT
            Not After : Oct  6 07:50:55 2019 GMT
        Subject: O=Allied-Telesis, CN=Tom
```

Table 73-3: Parameters in the output of the show crypto pki certificates user command

Parameter	Description
`Certificate`	Certificate name.
`Version`	Protocol version.
`Serial Number`	Serial number of the certificate.
`Signature Algorithm`	Algorithm used for the certificate signature.
`Issuer`	Subject of issuer creating the certificate.
`Validity`	Validity period.
`Subject`	Subject of the certificate.

Related Commands **crypto pki enroll local user**

show crypto pki trustpoints

Use this command to display trustpoint information.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax

```
show crypto pki trustpoints
```

Mode

User Exec and Privileged Exec

Example

The following command displays trustpoint information.

```
awplus# show crypto pki trustpoint
```

Output

Figure 73-4: Example output from the show crypto pki trustpoints command

```
Trustpoint local:
Subject Name:
CN = AlliedwarePlusCA
o = Allied-Telesis
Serial Number:0C
```

Table 73-4: Parameters in the output of the show crypto pki trustpoints command

Parameter	Description
`Subject Name`	CA certificate subject.
`Serial Number`	Current serial number of CA.

Related Commands

crypto pki enroll local

show radius local-server group

Use this command to display information about the local RADIUS server user group.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show radius local-server group [<user-group-name>]`

Parameter	Description
<user-group-name>	User group name string.

Mode User Exec and Privileged Exec

Example The following command displays Local RADIUS server user group information.

```
awplus# show radius local-server group
```

Output

Figure 73-5: Example output from the show radius local-server group command

```
Group-Name          Vlan
-------------------------------------------------------------------
NetworkOperators    ManagementNet
NormalUsers         CommonNet
```

Table 73-5: Parameters in the output of the show radius local-server group command

Parameter	Description
`Group-Name`	Group name.
`Vlan`	VLAN name assigned to the group.

Related Commands group

show radius local-server nas

Use this command to display information about NAS (Network Access Servers) registered to the local RADIUS server.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show radius local-server nas [<ip-address>]`

Parameter	Description
`<ip-address>`	Specify NAS IP address for show output.

Mode User Exec and Privileged Exec

Example The following command displays NAS information.

```
awplus# show radius local-server nas
```

Output

Figure 73-6: Example output from the show radius local-server nas command

```
NAS-Address     Shared-Key
-----------------------------------------------------------
127.0.0.1       awplus-local-radius-server
```

Table 73-6: Parameters in the output of the show radius local-server nas command

Parameter	Description
`NAS-Address`	IP address of NAS.
`Shared-Key`	Shared key used for RADIUS connection.

Related Commands **nas**

show radius local-server statistics

Use this command to display statistics about the local RADIUS server.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show radius local-server statistics
```

Mode

User Exec and Privileged Exec

Usage

Both unknown usernames and invalid passwords will display as failed logins in the show output.

Example

The following command displays Local RADIUS server statistics.

```
awplus# show radius local-server statistics
```

Output

Figure 73-7: Example output from the show radius local-server statistics command

```
Server status  : Run (administrative status is enable)
Enabled methods: MAC EAP-MD5 EAP-TLS EAP-PEAP

Successes               :1  Unknown NAS              :0
Failed Logins           :0  Invalid packet from NAS  :0
Internal Error          :0  Unknown Error            :0

NAS : 127.0.0.1
Successes               :0  Shared key mismatch      :0
Failed Logins           :0  Unknown RADIUS message   :0
Unknown EAP message     :0  Unknown EAP auth type    :0
Corrupted packet        :0

NAS : 192.168.1.61
Successes               :0  Shared key mismatch      :0
Failed Logins           :0  Unknown RADIUS message   :0
Unknown EAP message     :0  Unknown EAP auth type    :0
Corrupted packet        :0

NAS : 192.168.1.63
Successes               :1  Shared key mismatch      :0
Failed Logins           :0  Unknown RADIUS message   :0
Unknown EAP message     :0  Unknown EAP auth type    :0
Corrupted packet        :0

NAS : 192.168.1.65
Successes               :0  Shared key mismatch      :0
Failed Logins           :0  Unknown RADIUS message   :0
Unknown EAP message     :0  Unknown EAP auth type    :0
Corrupted packet        :0

Username          Successes  Failures
a                 1          0
admin             0          0
```

Related Commands

clear radius local-server statistics
radius-server local
server enable
server auth-port

show radius local-server user

Use this command to display information about the local RADIUS server user.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax

```
show radius local-server user [<user-name>]
show radius local-server user <user-name> format csv
```

Parameter	Description
`<user-name>`	RADIUS user name. If no user name is specified, information for all users is displayed.
`format`	File format.
`csv`	Comma separated value format.

Mode User Exec and Privileged Exec

Usage You can send output from any show command, including the CSV format output from this command, to a file. See "Controlling "show" Command Output" on page 1.37.

Examples The following command displays Local RADIUS server user information for user Tom.

```
awplus# show radius local-server user Tom
```

Figure 73-8: Example output from the show radius local-server user command

```
User-Name   Password    Group          Vlan
-----------------------------------------------------------------
Tom         abcd        NetworkOperators ManagementNet
```

The following command displays all Local RADIUS server information for all users.

```
awplus# show radius local-server user
```

The following command displays Local RADIUS server user information for TOM in CSV format.

```
awplus# show radius local-server user Tom format csv
```

Figure 73-9: Example output from the show radius local-server user csv command

```
true,"NetworkOperators","Tom",
"abcd",0,2099/01/
01,1,"","","ManagementNet",false,3600,false,0,"",false,"
```

Table 73-7: Parameters in the output from the show radius local-server user command

Parameter	Description
`User-Name`	User name.
`Password`	User password.
`Group`	Group name assigned to the user.
`Vlan`	VLAN name assigned to the user.

Related Commands

group
user (RADIUS server)

user (RADIUS server)

Use this command to register a user to the local RADIUS server.

Use the **no** variant of this command to delete a user from the local RADIUS server.

Syntax

```
user <radius-user-name> [encrypted] password <user-password>
    [group <user-group>]

no user <radius-user-name>
```

Parameter	Description
<radius-user-name>	RADIUS user name. This can also be a MAC address in the IEEE standard format of HH-HH-HH-HH-HH-HH if you are configuring MAC authentication to use local RADIUS server.
encrypted	Specifies that the password is being entered in its encrypted form, so that it is not further encrypted. When creating a new user, enter the password in plaintext, and do not use the **encrypted** parameter. Use the **encrypted** parameter only when referring to a user that has previously been created. For instance, when adding an existing user from another RADIUS server, use the **encrypted** parameter, and enter the encrypted version of the password that appears in the output of **show** commands for the user.
<user-password>	User password. This can also be a MAC address in the IEEE standard format of HH-HH-HH-HH-HH-HH if you are configuring MAC authentication to use local RADIUS server.
group	Specify the group for the user.
<user-group>	User group name.

Mode RADIUS Server Configuration

Usage RADIUS user names cannot contain question mark (?), space ( ), or quote (“ “) characters. RADIUS user names containing the below characters cannot use certificate authentication:

```
/ \ ‘ $ & ( ) * ; < > ‘ |
```

Certificates cannot be created and exported for RADIUS user names that contain the above characters. We advise you to avoid using these characters in RADIUS user names if you need to use certificate authentication, because you will not be able to create and export certificates.

You also can use the IEEE standard format hexadecimal notation (HH-HH-HH-HH-HH-HH) to specify a supplicant MAC address to configure the user name and user password parameters to use local RADIUS server for MAC Authentication. See the **Sample MAC-Authentication Configuration** in **Chapter 66, AAA Introduction and Configuration**. See also the command **user 00-db-59-ab-70-37 password 00-db-59-ab-70-37** as shown in the command examples.

Examples

The following commands add user `Tom` to the local RADIUS server and sets his password to `QwerSD`.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# user Tom password QwerSD
```

The following commands add user `Tom` to the local RADIUS server user group NormalUsers and sets his password `QwerSD`.

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# user Tom password QwerSD group
NormalUsers
```

The following commands remove user Tom from the local RADIUS server:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no user Tom
```

The following commands add the supplicant MAC address 00-d0-59-ab-70-37 to the local RADIUS server:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# user 00-db-59-ab-70-37 password 00-db-
59-ab-70-37
```

The following commands remove the supplicant MAC address 00-d0-59-ab-70-37 from the local RADIUS server:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# no user 00-db-59-ab-70-37
```

Related Commands

group
show radius local-server user

vlan (RADIUS server)

Use this command to set the VLAN ID or name for the local RADIUS server user group. The VLAN information is used for authentication with the dynamic VLAN feature.

Use the **no** variant of this command to clear the VLAN ID or VLAN name for the local RADIUS server user group.

Syntax

```
vlan {<vid>|<vlan-name>}
no vlan
```

Parameter	Description
<vid>	VLAN ID.
<vlan-name>	VLAN name.

Default VLAN information is not set by default.

Mode RADIUS Server Group Configuration

Examples The following commands set VLAN ID 200 to the group named NormalUsers:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# vlan 200
```

The following commands remove VLAN ID 200 from the group named NormalUsers:

```
awplus# configure terminal
awplus(config)# radius-server local
awplus(config-radsrv)# group NormalUsers
awplus(config-radsrv-group)# no vlan
```

Related Commands

group
show radius local-server user

Chapter 74: TACACS+ Introduction and Configuration

Introduction 74.2
TACACS+ Overview 74.2
The AlliedWare Plus TACACS+ Implementation 74.2
Authentication 74.3
Authorization 74.3
Accounting 74.4
Configuration 74.5
Configure TACACS+ 74.5
TACACS+ Configuration Example 74.7

Introduction

This chapter provides information about the AlliedWare Plus implementation of TACACS+ and how to configure it on this switch. For detailed descriptions of the commands used to configure TACACS+, see **Chapter 75, TACACS+ Commands**. For information about Authentication, Authorization and Accounting (AAA), see **Chapter 66, AAA Introduction and Configuration** and **Chapter 67, AAA Commands**.

TACACS+ Overview

TACACS+ (Terminal Access Controller Access-Control System Plus) provides a method for securely managing multiple network access points from a single management service.

TACACS+ is a TCP-based access control protocol, utilizing TCP port 49, that allows a device to forward a user's username and password to an authentication server to determine whether access can be allowed. In addition to this authentication service, TACACS+ can also provide authorization and accounting services.

One of the features of TACACS+ is the ability to separate authentication, authorization and accounting so that these functions can be provided independently on separate servers. Authentication involves identifying a user, typically by requiring the user to supply a valid username and password before access is granted. Following authentication, the user must gain authorization to perform tasks. For example, after logging into a switch, a user may try to issue configuration commands. The authorization process determines whether the user has the authority to issue these commands. Authorization is always preceded by authentication.

The AlliedWare Plus TACACS+ Implementation

The AlliedWare Plus TACACS+ implementation provides authentication, authorization, and accounting. Note that:

- Authorization cannot be performed independently of the authentication process. There are no authorization commands available.
- Authentication and authorization must be configured on the same server.
- Authorization is only applicable if enable password authentication has not been configured with the **aaa authentication enable default group tacacs+** command.

With the AlliedWare Plus TACACS+ implementation, all traffic that passes between the TACACS+ client and the TACACS+ servers on the network is encrypted. TACACS+ encrypts the entire payload of packets, which means that it encrypts the user's password between the client and the server.

A TACACS+ client is available on your switch. You need a system running TACACS+ server software from a software provider to use the TACACS+ functionality on your switch.

Authentication

The TACACS+ protocol can forward many types of username and password information. The AlliedWare Plus TACACS+ implementation supports username and password login authentication, as well as enable password authentication. This information is encrypted over the network with MD5 (Message Digest 5).

When TACACS+ login authentication is enabled on the switch with the **aaa authentication login** command and at least one TACACS+ server is configured and reachable, all user login authentications are authenticated against the TACACS+ server. No local login or other means of authentication is allowed or accepted by the switch unless the switch has been configured to use another authentication method as a backup, and the TACACS+ server is not reachable.

When TACACS+ enable password authentication is enabled on the switch with the **aaa authentication enable default group tacacs+** command and at least one TACACS+ server is configured and reachable, all user attempts to access a higher privilege level using the **enable (Privileged Exec mode)** command are authenticated against the TACACS+ server. If TACACS+ enable password authentication is enabled and the TACACS+ server is not reachable, then the user is only granted access to the desired privilege level if a backup authentication method is also configured.

Note

If TACACS+ login authentication is enabled on the switch, and enable password authentication is configured as default with the **aaa authentication enable default local** command, then a local enable password must be configured for each privilege level that needs to be accessible to users.

Authorization

In the AlliedWare Plus TACACS+ implementation, authorization cannot be performed independently of the authentication process. Authorization is concerned with what users are allowed to do once they have gained access to the managed device. This involves the passing of Attribute Value pairs (AV pairs) from the TACACS+ server to the managed device. An AV pair is made up of two pieces of information: the attribute that identifies the parameter to be set, and the value that specifies the value to assign to that parameter. These AV pairs are configured on a per-user or per-group basis on the TACACS+ server. The AV pairs that are supported by the AlliedWare Plus TACACS+ implementation are:

- **Privilege Level**

  Privilege levels range from 1 to 15, with 15 being the highest. For information about privilege levels see **"How to Add and Remove Users" on page 1.27** and the **username** command on page 5.38.

- **Timeout**

  The value assigned to this attribute specifies the length of time that the session can exist. After this value has expired, the session will either be disconnected, or have the privilege of the user reduced. The valid range of timeout values is 0 to 65535 (minutes).

- **Idletime**

  If no input or output traffic is received or sent in the period specified by the value for this attribute, the session is disconnected. The valid idletime range is 0 to 65535 (minutes).

Note

In the AlliedWare Plus TACACS+ implementation, authorization for privilege level, timeout, and idletime AV pairs is only attempted if enable password authentication (**aaa authentication enable default group tacacs+** command) is not configured. If enable password authentication is configured then the privilege level a user is granted access to is determined during the enable password authentication session.

Accounting

TACACS+ accounting usually takes place after authentication and authorization. However, because TACACS+ separates these three functions, neither authentication nor authorization are required for accounting to function. TACACS+ accounting provides the following two distinct functions:

- a record of services used for billing purposes
- an audit trail for user exec sessions

The AlliedWare Plus TACACS+ accounting implementation supports an audit trail for user exec sessions only. This includes the ability to configure accounting for user logins and logouts, and accounting of any commands executed by the user while they are logged into the switch.

TACACS+ accounting includes three different types of accounting records:

- **start** records that indicate a service is about to start
- **stop** records that indicate a service has just ended
- **update** records that indicate a service is still in progress

Configuration

This section describes how to set up TACACS+ for login authentication, enable password authentication, and accounting.

The TACACS+ server is normally a multiuser system running TACACS+ server software from a software provider. TACACS+ servers are identified on the basis of their host name or IP address. A TACACS+ server and a switch use a shared secret text string to encrypt passwords and exchange responses. To configure TACACS+, you must specify the host running the TACACS+ server software and a secret text string that it shares with the switch.

Configure TACACS+

Table 74-1: General configuration procedure for TACACS+ authentication and accounting

Step 1. Specify a remote TACACS+ server and the shared secret key	
`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`tacacs-server host {<host-name>|<ip-address>} [key [8]<key-string>]`	Specify the IP address or host name of the remote TACACS+ server host and the shared secret key to use with the specified TACACS+ server. Specify 8 if you are entering a password as a string that has already been encrypted instead of entering a plain text password. As many as four TACACS+ servers can be configured and consulted for authentication and accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn.
`awplus(config)#`	
`tacacs-server key [8] <key-string>`	Specify the global shared secret text string used between the switch and **all** TACACS+ servers. Specify `8` if you are entering a password as a string that has already been encrypted instead of entering a plain text password. If no secret key is explicitly specified for a TACACS+ server with the **tacacs-server host** command, the global secret key will be used.
Step 2. Specify the timeout value	
`awplus(config)#`	
`tacacs-server timeout <seconds>`	Specify for how many seconds a switch waits for a reply to a TACACS+ request before considering the TACACS+ server dead.

Table 74-1: General configuration procedure for TACACS+ authentication and accounting(cont.)

Step 3. Define the method list for TACACS+ login authentication	
`awplus(config)#`	
`aaa authentication login {default|<list-name>} {[local] [group {radius|tacacs+|<group-name>}]}`	This method list defines the AAA server type used for login authentication. The server types are always used in the order specified with this command. If the first server in the method list is unreachable, the switch sends the request to the next server in the list. If the authentication server denies the authentication request because of an incorrect username or password then the user login fails.
Step 4. Define the method list for TACACS+ enable password authentication	
`awplus(config)#`	
`aaa authentication enable default group tacacs+ [local] [none]`	This method list defines the authentication method used to determine the privilege command level a user can access. Specify **local** to use the locally configured enable password and **none** to grant access to Privileged Exec mode with no authentication, if the TACACS+ server goes offline, or is not reachable during enable password authentication.
Step 5. Define the method for TACACS+ login accounting	
`awplus(config)#`	
`aaa accounting login {default|<list-name>} {start-stop|stop-only|none} {group {radius|tacacs+|<group-name>}}`	You can only define one method for login accounting, either RADIUS or TACACS+. Specify **start-stop** to send both start and stop login accounting records, **stop-only** to send only stop login accounting records, or **none** to disable the sending of login accounting records.
Step 6. Configure TACACS+ command accounting	
`awplus(config)#`	
`aaa accounting commands <1-15> default stop-only group tacacs+`	TACACS+ command accounting is configured per privilege level and only commands of the specified privilege level are accounted. Therefore, if you require that all commands are accounted to the TACACS+ server, you must configure command accounting for each privilege level separately. Commands are accounted to the TACACS+ server after they have successfully executed.
Step 7. Troubleshooting TACACS+	
`awplus(config)#`	
`show tacacs+`	Display the current TACACS+ server configuration and status.
`awplus#`	
`debug aaa authentication`	Enable debug output for TACACS+ authentication.
`awplus#`	
`debug aaa authorization`	Enable debug output for TACACS+ authorization.
`awplus#`	
`debug aaa accounting`	Enable debug output for TACACS+ accounting.

TACACS+ Configuration Example

Example The following example shows how to configure the switch to authenticate and account using TACACS+.

Figure 74-1: Sample TACACS+ authentication and accounting to configure the switch to authenticate and account user exec sessions

```
!
tacacs-server host 172.10.10.1
tacacs-server key tacacspass
aaa authentication login admin group tacacs+ local
aaa authentication enable default group tacacs+ local
aaa accounting login admin start-stop group tacacs+
aaa accounting commands 1 default stop-only group tacacs+
aaa accounting commands 7 default stop-only group tacacs+
aaa accounting commands 15 default stop-only group tacacs+

line console 0
login authentication admin
accounting login admin
!
```

The lines in this example TACACS+ authentication and accounting configuration are defined as follows:

- The **tacacs-server host** command defines the IP address of the TACACS+ server host.
- The **tacacs-server key** command defines the global shared secret text string between the network access server and the TACACS+ server host.
- The **aaa authentication login** command defines a method list named **admin** to use first the TACACS+ servers and then the local user database for user login authentication.
- The **aaa authentication enable default group tacacs+** command defines a method list to use first the TACACS+ servers and then the local enable passwords, set with the **enable password** command, for user enable password authentication.
- The **aaa accounting login** command defines a method named **admin** to use TACACS+ servers for login accounting.
- The **aaa accounting commands** command specifies the privilege level of the commands that will be accounted.
- The **login authentication** command specifies that this method list will be used for authenticating users logging in on the asynchronous console port.
- The **accounting login** command specifies that this method list will be used for accounting users logging in on the asynchronous console port.

Chapter 75: TACACS+ Commands

Command List 75.2
show tacacs+ 75.2
tacacs-server host 75.3
tacacs-server key 75.5
tacacs-server timeout 75.6

Command List

This chapter provides an alphabetical reference for commands used to configure the device to use TACACS+ servers. For more information about TACAC+, see **Chapter 74, TACACS+ Introduction and Configuration**.

show tacacs+

This command displays the current TACACS+ server configuration and status.

Syntax `show tacacs+`

Mode User Exec and Privileged Exec

Example To display the current status of TACACS+ servers, use the command:

```
awplus# show tacacs+
```

Output **Figure 75-1: Example output from the show tacacs+ command**

```
TACACS+ Global Configuration
  Timeout              : 5 sec

Server Host/      Server
IP Address        Status
------------------------------------------------------------
192.168.1.10      Alive
192.168.1.11      Unknown
```

Table 75-1: Parameters in the output of the show tacacs+ command

Output Parameter	Meaning	
`Timeout`	A time interval in seconds.	
`Server Host/IP Address`	TACACS+ server hostname or IP address.	
`Server Status`	The status of the authentication port.	
	`Alive`	The server is alive.
	`Dead`	The server has timed out.
	`Error`	The server is not responding or there is an error in the key string entered.
	`Unknown`	The server is never used or the status is unkown.
	`Unreachable`	The server is unreachable.
	`Unresolved`	The server name can not be resolved.

tacacs-server host

Use this command to specify a remote TACACS+ server host for authentication, authorization and accounting, and to set the shared secret key to use with the TACACS+ server. The parameters specified with this command override the corresponding global parameters for TACACS+ servers.

Use the **no** variant of this command to remove the specified server host as a TACACS+ authentication and authorization server.

Syntax

```
tacacs-server host {<host-name>|<ip-address>} [key [8]<key-string>]
no tacacs-server host {<host-name>|<ip-address>}
```

Parameter	Description
<host-name>	Server host name. The DNS name of the TACACS+ server host.
<ip-address>	The IP address of the TACACS+ server host, in dotted decimal notation A.B.C.D.
`key`	Set shared secret key with TACACS+ servers.
`8`	Specifies that you are entering a password as a string that has already been encrypted instead of entering a plain text password. The running config displays the new password as an encrypted string even if password encryption is turned off.
<key-string>	Shared key string applied, a value in the range 1 to 64 characters. Specifies the shared secret authentication or encryption key for all TACACS+ communications between this device and the TACACS+ server. This key must match the encryption used on the TACACS+ server. This setting overrides the global setting of the **tacacs-server key** command. If no key value is specified, the global value is used.

Default No TACACS+ server is configured by default.

Mode Global Configuration

Usage A TACACS+ server host cannot be configured multiple times like a RADIUS server.

As many as four TACACS+ servers can be configured and consulted for login authentication, enable password authentication and accounting. The first server configured is regarded as the primary server and if the primary server fails then the backup servers are consulted in turn. A backup server is consulted if the primary server fails, not if a login authentication attempt is rejected. The reasons a server would fail are:

- it is not network reachable
- it is not currently TACACS+ capable
- it cannot communicate with the switch properly due to the switch and the server having different secret keys

Examples

To add the server `tac1.company.com` as the TACACS+ server host, use the following commands:

```
awplus# configure terminal
awplus(config)# tacacs-server host tac1.company.com
```

To set the secret key to `secret` on the TACACS+ server `192.168.1.1`, use the following commands:

```
awplus# configure terminal
awplus(config)# tacacs-server host 192.168.1.1 key secret
```

To remove the TACACS+ server `tac1.company.com`, use the following commands:

```
awplus# configure terminal
awplus(config)# no tacacs-server host tac1.company.com
```

Related Commands

aaa accounting commands
aaa authentication login
tacacs-server key
tacacs-server timeout
show tacacs+

tacacs-server key

This command sets a global secret key for TACACS+ authentication, authorization and accounting. The shared secret text string is used for TACACS+ communications between the switch and all TACACS+ servers.

Note that if no secret key is explicitly specified for a TACACS+ server with the **tacacs-server host** command, the global secret key will be used for the shared secret for the server.

Use the **no** variant of this command to remove the global secret key.

Syntax

```
tacacs-server key [8] <key-string>
no tacacs-server key
```

Parameter	Description
8	Specifies a string in an encrypted format instead of plain text. The running config will display the new password as an encrypted string even if password encryption is turned off.
<key-string>	Shared key string applied, a value in the range 1 to 64 characters. Specifies the shared secret authentication or encryption key for all TACACS+ communications between this device and all TACACS+ servers. This key must match the encryption used on the TACACS+ server.

Mode Global Configuration

Usage Use this command to set the global secret key shared between this client and its TACACS+ servers. If no secret key is specified for a particular TACACS+ server using the **tacacs-server host** command, this global key is used.

Examples To set the global secret key to `secret` for TACACS+ server, use the following commands:

```
awplus# configure terminal
awplus(config)# tacacs-server key secret
```

To delete the global secret key for TACACS+ server, use the following commands:

```
awplus# configure terminal
awplus(config)# no tacacs-server key
```

Related Commands **tacacs-server host**
show tacacs+

tacacs-server timeout

Use this command to specify the TACACS+ global timeout value. The timeout value is how long the device waits for a reply to a TACACS+ request before considering the server to be dead.

Note that this command configures the **timeout** parameter for TACACS+ servers globally.

The **no** variant of this command resets the transmit timeout to the default (5 seconds).

Syntax

```
tacacs-server timeout <seconds>
no tacacs-server timeout
```

Parameter	Description
<seconds>	TACACS+ server timeout in seconds, in the range 1 to 1000.

Default The default timeout value is 5 seconds.

Mode Global Configuration

Examples To set the timeout value to 3 seconds, use the following commands:

```
awplus# configure terminal
awplus(config)# tacacs-server timeout 3
```

To reset the timeout period for TACACS+ servers to the default, use the following commands:

```
awplus# configure terminal
awplus(config)# no tacacs-server timeout
```

Related Commands **tacacs-server host**
show tacacs+

Chapter 76: Secure Shell (SSH) Introduction

Introduction 76.2
Secure Shell on the AlliedWare Plus OS 76.2
Configuring the SSH Server 76.4
Creating a Host Key 76.4
Enabling the Server 76.4
Modifying the Server 76.5
Validating the Server Configuration 76.6
Adding SSH Users 76.6
Authenticating SSH Users 76.7
Adding a Login Banner 76.7
Monitoring the Server and Managing Sessions 76.8
Debugging the Server 76.8
Configuring the SSH Client 76.9
Modifying the Client 76.9
Adding SSH Servers 76.10
Authenticating with a Server 76.10
Connecting to a Server and Running Commands 76.11
Copying files to and from the Server 76.11
Debugging the Client 76.11

Introduction

This chapter describes how the Secure Shell protocol is implemented in the AlliedWare Plus™ Operating System. It covers:

- Support for Secure Shell.
- Configuring your device as a Secure Shell server and client.
- Using Secure Shell to manage your device.

The AlliedWare Plus™ OS supports SSH version 2 and SSH version 1.5, making it backwards compatible with SSH version 1.

Secure management is important in modern networks, as the ability to easily and effectively manage switches and routers, and the requirement for security, are two almost universal requirements. Protocols such as Telnet and rlogin allow you to manage devices remotely, but can have serious security problems, such as relying on reusable plaintext passwords that are vulnerable to wiretapping or password guessing. The Secure Shell (SSH) protocol is superior to these protocols by providing encrypted and strongly authenticated remote login sessions.

SSH provides sessions between a host running a SSH server and a machine with a SSH client. The AlliedWare Plus™ OS includes both a SSH server and a SSH client to enable you to securely—with the benefit of cryptographic authentication and encryption—manage your devices over an insecure network:

- SSH replaces Telnet for remote terminal sessions; SSH is strongly authenticated and encrypted.
- Remote command execution allows you to send commands to a device securely and conveniently, without requiring a terminal session on the device.
- SSH allows you to connect to another host from your switch or router.

The AlliedWare Plus™ OS supports Secure Copy (SCP) and SSH File Transfer Protocol (SFTP). Both these protocols allow you to securely copy files between your device and remote machines. SFTP provides additional features from SCP, such as allowing you to manipulate the remote files, and halt or resume file transfers without closing the session.

Secure Shell on the AlliedWare Plus OS

The AlliedWare Plus™ OS implementation of SSH is compatible with the following RFCs and Internet Drafts:

- The Secure Shell (SSH) Protocol Architecture (RFC 4251)
- The Secure Shell (SSH) Authentication Protocol (RFC 4252)
- The Secure Shell (SSH) Transport Layer Protocol (RFC 4253)
- The Secure Shell (SSH) Connection Protocol (RFC 4254)
- The SSH (Secure Shell) Remote Login Protocol (draft-ylonen-ssh-protocol-00.txt)
- SSH File Transfer Protocol (draft-ietf-secsh-filexfer-13.txt)

Secure Shell supports the following features for both SSH version 2 and SSH version 1.5:

- Inbound SSH connections (server mode) and outbound SSH connections (client mode).
- File loading to and from remote machines using Secure Copy, using either the SSH client or SSH server mode.
- RSA public keys with lengths of 768–32768 bits, and DSA keys with lengths of 1024 bits. Keys are stored in a format compatible with other SSH implementations, and mechanisms are provided to copy keys to and from your device.
- Secure encryption, such as Triple DES and Blowfish.
- Remote non-interactive shell that allows arbitrary commands to be sent securely to your device, possibly automatically.
- Compression of Secure Shell traffic.
- Tunneling of TCP/IP traffic.

Secure Shell supports the following features for SSH version 2 only:

- File loading from remote machines using SSH File Transfer Protocol (SFTP).
- A login banner on the SSH server, that displays when SSHv2 clients connect to the server.

Configuring the SSH Server

This section provides instructions on:

- Creating a Host Key
- Enabling the Server
- Modifying the Server
- Validating the Server Configuration
- Adding SSH Users
- Authenticating SSH Users
- Adding a Login Banner
- Monitoring the Server and Managing Sessions
- Debugging the Server

Creating a Host Key

The SSH server uses either an RSA or DSA host key to authenticate itself with SSH clients. This key must be configured before the SSH server can operate. If no host key exists, you cannot start the SSH server.

Once created, the host key is stored securely on the device. To generate a host key for the SSH server, use the command:

```
awplus(config)# crypto key generate hostkey {dsa|rsa|rsa1}
                [<768-32768>]
```

This command has two parameters for creating RSA keys. The **rsa** parameter creates a host key for SSH version 2 sessions only. To create a host key for SSH version 1 sessions, use the **rsa1** parameter.

To destroy a host key, use the command:

```
awplus(config)# crypto key destroy hostkey {dsa|rsa|rsa1}
```

To display a host key stored on your device, use the command:

```
awplus(config)# show crypto key hostkey [dsa|rsa|rsa1]
```

Enabling the Server

You must enable the SSH server before connections from SSH, SCP, and SFTP clients are accepted. When the SSH server is disabled it rejects connections from SSH clients. The SSH server is disabled by default on your device.

To enable the SSH server, use the command:

```
awplus(config)# service ssh [ip|ipv6]
```

To disable the SSH server, use the command:

```
awplus(config)# no service ssh [ip|ipv6]
```

When enabled, the SSH server allows SCP and SFTP sessions by default. To disable these services, use the commands:

```
awplus(config)# no ssh server scp

awplus(config)# no ssh server sftp
```

This allows you to reject SCP or SFTP file transfer requests, while still allowing Secure Shell connections. To re-enable SCP and SFTP services, use the command:

```
awplus(config)# ssh server scp

awplus(config)# ssh server sftp
```

Modifying the Server

To modify the SSH version that the server supports, or the TCP port that the server listens to for incoming sessions, use the command:

```
awplus(config)# ssh server {[v1v2|v2only]|<1-65535>}
```

The server listens on port 22 for incoming sessions, and supports both SSH version 2 and SSH version 1, by default.

To modify session and login timeouts on the SSH server, and the number of unauthenticated connections the server allows, use the command:

```
awplus(config)# ssh server {[session-timeout <0-3600>]
               [login-timeout <1-600>]
               [max-startups <1-128>]}
```

The SSH server waits 60 seconds for a client to authenticate itself, by default. You can alter this waiting time by using the **login-timeout** parameter. If the client is still not authenticated after the set timeout, then the SSH server disconnects the session.

The SSH server only allows only 10 unauthenticated SSH sessions at any point in time, by default. You can modify the number of unauthenticated sessions it allows, by using the **max-startups** parameter.

Once a client has authenticated, the SSH session does not time out, by default. Use the **session-timeout** parameter to set a **maximum time period the server waits before deciding that a session is inactive and terminating it**

For example, to set the session timeout to 600 seconds, the login timeout to 30 seconds, and the maximum number of concurrent unauthenticated sessions to 5, use the command:

```
awplus(config)# ssh server session-timeout 600 login-timeout
               30 max-startups 5
```

To remove the configured session timeout, login timeout, or maximum startups, use the command:

```
awplus(config)# no ssh server session-timeout login-timeout
               max-startups
```

Validating the Server Configuration

To validate the SSH server configuration, use the command:

```
awplus(config)# show running-config ssh
```

Adding SSH Users

The SSH server requires you to register SSH users. Users that are not registered cannot access the SSH server. Ensure first that you have defined the user in the Authorized User Database of your device. To add a new user, use the command:

```
awplus(config)# username USERNAME (privilege 1-15) password
                PASSWORD
```

To register a user with the SSH server, use the command:

```
awplus(config)# ssh server allow-users <username-pattern>
                [<hostname-pattern>]
```

Registered entries can contain just the username, or the username with some host details, such as an IP address range. Additionally you can specify a range of users or hostname details by using an asterisk to match any string of characters. For example, to allow any user from the IP range 192.168.1.1 to 192.168.1.255, use the command:

```
awplus(config)# ssh server allow-users * 192.168.1.*
```

To display the list of allowed users, use the command:

```
awplus# show ssh server allow-users
```

To delete an entry from the list of allowed users, use the command:

```
awplus(config)# no ssh server allow-users <username-pattern>
                [<hostname-pattern>]
```

The SSH server also contains a list of denied users. The server checks all incoming sessions against this list and denies any matching session, regardless of whether the session matches an entry in the allowed users list. To add an entry to the list of denied users, use the command:

```
awplus(config)# ssh server deny-users <username-pattern>
                [<hostname-pattern>]
```

This allows you to deny specific users from a range of allowed users. For example, to deny a user with the IP address 192.168.1.12, use the command:

```
awplus(config)# ssh server deny-users * 192.168.1.12
```

To display the database of denied users, use the command:

```
awplus# show ssh server deny-users
```

To delete a client from the database of denied users, use the command:

```
awplus(config)# no ssh server deny-users <username-pattern>
                [<hostname-pattern>]
```

Authenticating SSH Users

SSH users can use either their password or public key authentication to authenticate themselves with the SSH server. To use public key authentication, copy the user's public key file from their client device to the SSH server. To associate the key with a user, use the command:

```
awplus(config)# crypto key pubkey-chain userkey <username>
                [<filename>]
```

For example, to associate the file key.pub with the user "langley", use the command:

```
awplus(config)# crypto key pubkey-chain userkey langley
                key.pub
```

To add a key as text into the terminal for user "geoff", first enter the command:

```
awplus(config)# crypto key pubkey-chain userkey geoff
```

then paste or type the key in as text.

You can add multiple keys for the same user. To display the list of public keys associated with a user, use the command:

```
awplus(config)# show crypto key pubkey-chain userkey
                <username> [<1-65535>]
```

The **<1-65535>** parameter allows you to display an individual key.

To delete a key associated with a user from your device, use the command:

```
awplus(config)# no crypto key pubkey-chain userkey
                <username> <1-65535>
```

Adding a Login Banner

You can add a login banner to the SSH server for sessions with SSH version 2 clients. The server displays the banner to clients before the login prompt. To set the login banner's message, use the command:

```
awplus(config)# banner login
```

then enter your message and use Ctrl+D to finish.

To view the configured login banner, use the command:

```
awplus# show banner login
```

To remove the configured message for the login banner, use the command:

```
awplus(config)# no banner login
```

Monitoring the Server and Managing Sessions

To display the current status of the SSH server, use the command:

```
awplus# show ssh server
```

To display the current status of SSH sessions on your device, use the command:

```
awplus# show ssh
```

Note that this displays both SSH server and SSH client sessions that your Allied Telesis device is running. Use this command to view the unique identification number assigned to each incoming or outgoing SSH session. You need the ID number when terminating a specific session from your device.

To terminate a session, or all sessions, use the command:

```
awplus# clear ssh {<1-65535>|all}
```

Debugging the Server

Information which may be useful for troubleshooting the SSH server is available using the SSH debugging function. You can enable server debugging while the SSH server is functioning. Use the command:

```
awplus# debug ssh server [brief|full]
```

To disable SSH server debugging, use the command:

```
awplus# no debug ssh server
```

Configuring the SSH Client

This section provides instructions on:

- **Modifying the Client**
- **Adding SSH Servers**
- **Authenticating with a Server**
- **Connecting to a Server and Running Commands**
- **Copying files to and from the Server**
- **Debugging the Client**

Modifying the Client

You can configure a selection of variables when using the SSH client. Note that the following configuration commands apply only to client sessions initiated after the command. The configured settings are not saved; after you have logged out from the SSH client, the client returns to using the default settings. Use the command:

```
awplus(config)# ssh client {port <1-65535>|version {1|2}|
                session-timeout <0-3600>|connect-timeout
                <1-600>}
```

The SSH client uses TCP port 22, by default. You can change the TCP port for the remote SSH server by using the **port** parameter.

The client supports both SSH version 1 and version 2 sessions, by default. To change the SSH client to only use a specific SSH version for sessions, for example SSH version 1, use the **version** parameter.

The client terminates sessions that are not established after 30 seconds, by default. You can change this time period by using the session-timeout parameter.

Once the client has authenticated with a server, the client does not time out the SSH session, by default. Use the **session-timeout** parameter to set a maximum time period the client waits before deciding that a session is inactive and terminating the session.

To modify the SSH client so that it uses port 2000 for sessions, and supports only SSH version 1 connections, use the command:

```
awplus(config# ssh client port 2000 version 1
```

To modify the SSH client so that unestablished sessions time out after 60 seconds, and inactive sessions time out after 100 seconds, use the command:

```
awplus(config)# ssh client session-timeout 100 connect-timeout
                100
```

To remove the configured port, SSH version, session timeout, and connection timeout settings, use the command:

```
awplus(config)# no ssh client port version session-timeout
                connect-timeout
```

Adding SSH Servers

SSH servers identify themselves using a host key (see **"Creating a Host Key" on page 76.4**). Before the SSH client establishes a session with a SSH server, it confirms that the host key sent by the server matches its database entry for the server. If the database does not contains a host key for the server, then the SSH client requires you to confirm that the host key sent from the server is correct.

To add an SSH server to the client's database, use the command:

```
awplus# crypto key pubkey-chain knownhosts [ip|ipv6]
       <hostname> [rsa|dsa|rsa1]
```

To display the SSH servers in the client's database, use the command:

```
awplus# show crypto key pubkey-chain knownhosts
       [<1-65535>]
```

To remove an entry in the database, use the command:

```
awplus# no crypto key pubkey-chain knownhosts <1-65535>
```

Authenticating with a Server

You can authenticate your session with a server by either using a password, or using RSA or DSA public key authentication. To use public key authentication, you must generate a pair of keys, one private and one public, and copy the public key onto the SSH server.

To generate an RSA or DSA set of private and public keys for an SSH user, use the command:

```
awplus(config)# crypto key generate userkey <username> {dsa|
                rsa|rsa1} [<768-32768>]
```

You can generate one key of each encryption type per user on your client. When authenticating with an SSH server that supports SSH version 1 only, you must use a key generated by the **rsa1** parameter.

To copy the public key onto the SSH server, you must display the key onscreen. To display the public key associated with a user, use the command:

```
awplus# show crypto key userkey <username> [dsa|rsa|
       rsa1]
```

To display the public keys set for other users, you must specify their username. Only users with the highest privilege setting can use this command to view the keys of other users.

To delete a public and private pair of keys associated with a user, use the command:

```
awplus(config)# crypto key destroy userkey <username> {dsa|rsa|
                rsa1}
```

Connecting to a Server and Running Commands

To connect to a remote SSH server and execute a command, use the command:

```
awplus# ssh [ip|ipv6][{[user <username>]|[port
        <1-65535>]|[version {1|2}]}] <hostname>
        [<line>]
```

By default, the SSH client attempts to use SSH version 2 with the SSH server. If this fails, the client uses SSH version 1.

For example, to connect to the SSH server at 192.168.1.2 as user “john”, and execute the command “show sys”, use the command:

```
awplus# ssh user john 192.168.1.2 “show sys”
```

Copying files to and from the Server

You can use either the SCP or SFTP client to transfer files from a remote SSH server. Use the command:

```
awplus# copy <source-url> <destination-url>
```

For example, to use SFTP to load a file from the SSH server 192.168.1.2, onto the flash memory of your device, use the command:

```
awplus# copy sftp://192.168.1.2/key.pub flash
```

To upload files to the SSH server, you must use SCP. For example, to upload the file bobskey.pub as the user “bob”, use the command:

```
awplus# copy flash:/bobskey.pub scp://bob@192.168.1.2
```

For more information see **Chapter 6, Creating and Managing Files**.

Debugging the Client

Information which may be useful for troubleshooting the SSH client is available using the SSH debugging function. You can enable client debugging while the SSH client is functioning. Use the command:

```
awplus# debug ssh client [brief|full]
```

To disable SSH client debugging, use the command:

```
awplus# no debug ssh client
```

Chapter 77: Secure Shell (SSH) Configuration

SSH Server Configuration Example 77.2

SSH Server Configuration Example

This chapter provides a Secure Shell server configuration example. For more information about the SSH server, see **Chapter 76, Secure Shell (SSH) Introduction**. For detailed information about the commands used to configure the SSH server, see **Chapter 78, Secure Shell (SSH) Commands**.

The following example configures a SSH server where:

- the SSH server uses RSA encryption
- the SSH server is compatible with both SSH version 1 and version 2 clients
- three SSH users are configured: Manager, John, and Asuka. “manager” can connect from only a defined range of hosts, while “john” and “asuka” can SSH from all hosts
- the SSH users use RSA private and public key authentication

This example shows how to create RSA encryption keys, configure the Secure Shell server, and register users to make Secure Shell connections to your device.

Step 1: Login as a highest Privileged User.

To create the keys and add users, you must login as a privileged user.

Step 2: Create encryption keys.

Two RSA private keys are required before enabling the Secure Shell server for each type of SSH version. Use the commands:

```
awplus# configure terminal
awplus(config)# crypto key generate hostkey rsa
awplus(config)# crypto key generate hostkey rsa1
awplus(config)# exit
```

To verify the key creation, use the command:

```
awplus# show crypto key hostkey
```

Step 3: Enable the Secure Shell server.

Enable Secure Shell on the device using the commands:

```
awplus# configure terminal
awplus(config)# service ssh
```

Modify the SSH server settings as desired. For example, to set the login-timeout to 60, and the session-timeout to 3600, use the commands:

```
awplus(config)# ssh server session-timeout 3600 login-timeout
60
```

To verify the server configuration, use the command:

```
awplus# show ssh
```

Step 4: Create SSH users.

In order to connect and execute commands, you must register users in the SSH user database, and in the User Authentication Database of the device.

To create the users `john` and `asuka` in the User Authentication Database, use the commands:

```
awplus# configure terminal
awplus(config)# username john privilege 15 password secret
awplus(config)# username asuka privilege 15 password
very secret
```

To register `john` and `asuka` as SSH clients, use the commands:

```
awplus(config)# ssh server allow-users john
awplus(config)# ssh server allow-users asuka
```

To register “manager” as an SSH client so that can only connect from the IP address 192.168.1.1, use the command:

```
awplus(config)# ssh server allow-users manager 192.168.1.1
```

Step 5: Set up authentication.

SSH users cannot connect unless the server can authenticate them. There are two ways to authenticate an SSH session: password authentication, and RSA or DSA private/public key authentication. When using password authentication, the user must supply their User Authentication Database password.

To use private/public key authentication, copy the public keys for each user onto the device. To copy the files onto flash from the key directory of an attached TFTP server, use the command:

```
awplus# copy tftp://key/john.pub flash:/john.pub
awplus# copy tftp://key/asuka.pub flash:/asuka.pub
```

To associate the key file with each user, use the command:

```
awplus# configure terminal
awplus(config)# crypto key pubkey-chain userkey john john.pub
awplus(config)# crypto key pubkey-chain userkey asuka
asuka.pub
awplus(config)# crypto key pubkey-chain userkey manager
manager.pub
```

Chapter 78: Secure Shell (SSH) Commands

Introduction 78.2
Command List 78.2
banner login (SSH) 78.2
clear ssh 78.3
crypto key destroy hostkey 78.4
crypto key destroy userkey 78.5
crypto key generate hostkey 78.6
crypto key generate userkey 78.7
crypto key pubkey-chain knownhosts 78.8
crypto key pubkey-chain userkey 78.10
debug ssh client 78.12
debug ssh server 78.13
service ssh 78.14
show banner login 78.16
show crypto key hostkey 78.17
show crypto key pubkey-chain knownhosts 78.18
show crypto key pubkey-chain userkey 78.19
show crypto key userkey 78.20
show running-config ssh 78.21
show ssh 78.22
show ssh client 78.23
show ssh server 78.24
show ssh server allow-users 78.25
show ssh server deny-users 78.26
ssh 78.27
ssh client 78.29
ssh server 78.31
ssh server allow-users 78.33
ssh server authentication 78.35
ssh server deny-users 78.37
ssh server resolve-host 78.38
ssh server scp 78.39
ssh server sftp 78.40
undebug ssh client 78.41
undebug ssh server 78.41

Introduction

This chapter provides an alphabetical reference for commands used to configure Secure Shell (SSH). For more information, see **Chapter 76, Secure Shell (SSH) Introduction**, and **Chapter 77, Secure Shell (SSH) Configuration**.

Command List

banner login (SSH)

This command configures a login banner on the SSH server. This displays a message on the remote terminal of the SSH client before the login prompt. SSH client version 1 does not support this banner.

To add a banner, first enter the command **banner login**, and hit [Enter]. Write your message. You can use any character and spaces. Use Ctrl+D at the end of your message to save the text and re-enter the normal command line mode.

The banner message is preserved if the device restarts.

The **no** variant of this command deletes the login banner from the device.

Syntax

```
banner login
no banner login
```

Default No banner is defined by default.

Mode Global Configuration

Examples To set a login banner message, use the commands:

```
awplus# configure terminal
awplus(config)# banner login

Type CNTL/D to finish.
... banner message comes here ...
^D
awplus(config)#
```

and enter the message. Use Ctrl+D to finish.

To remove the login banner message, use the commands:

```
awplus# configure terminal
awplus(config)# no banner login
```

Related Commands **show banner login**

clear ssh

This command deletes Secure Shell sessions currently active on the device. This includes both incoming and outgoing sessions. The deleted sessions are closed. You can only delete an SSH session if you are a system manager or the user who initiated the session. If **all** is specified then all active SSH sessions are deleted.

Syntax `clear ssh {<1-65535>|all}`

Parameters	Description
`<1-65535>`	Specify a session ID in the range 1 to 65535 to delete a specific session.
`all`	Delete all SSH sessions.

Mode Privileged Exec

Examples To stop the current SSH session 123, use the command:

```
awplus# clear ssh 123
```

To stop all SSH sessions active on the device, use the command:

```
awplus# clear ssh all
```

Related Commands **service ssh**
ssh

crypto key destroy hostkey

This command deletes the existing public and private keys of the SSH server. Note that for an SSH server to operate it needs at least one set of hostkeys configured before an SSH server is started.

Syntax `crypto key destroy hostkey {dsa|rsa|rsa1}`

Parameters	Description
`dsa`	Deletes the existing DSA public and private keys.
`rsa`	Deletes the existing RSA public and private keys configured for SSH version 2 connections.
`rsa1`	Deletes the existing RSA public and private keys configured for SSH version 1 connections.

Mode Global Configuration

Example To destroy the RSA host key used for SSH version 2 connections, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key destroy hostkey rsa
```

Related Commands **crypto key generate hostkey**
service ssh

crypto key destroy userkey

This command destroys the existing public and private keys of an SSH user configured on the device.

Syntax `crypto key destroy userkey <username> {dsa|rsa|rsa1}`

Parameters	Description
`<username>`	Name of the user whose userkey you are destroying. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
`dsa`	Deletes the existing DSA userkey.
`rsa`	Deletes the existing RSA userkey configured for SSH version 2 connections.
`rsa1`	Deletes the existing RSA userkey for SSH version 1 connections.

Mode Global Configuration

Example To destroy the RSA user key for the SSH user `remoteuser`, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key destroy userkey remoteuser rsa
```

Related Commands
crypto key generate hostkey
show ssh
show crypto key hostkey

crypto key generate hostkey

This command generates public and private keys for the SSH server using either an RSA or DSA cryptography algorithm. You must define a host key before enabling the SSH server. Start SSH server using the **service ssh** command. If a host key exists with the same cryptography algorithm, this command replaces the old host key with the new key.

This command is not saved in the device configuration. However, the device saves the keys generated by this command in the non-volatile memory.

Syntax `crypto key generate hostkey {dsa|rsa|rsa1} [<768-32768>]`

Parameters	Description
`dsa`	Creates a DSA hostkey. Both SSH version 1 and 2 connections can use the DSA hostkey.
`rsa`	Creates an RSA hostkey for SSH version 2 connections.
`rsa1`	Creates an RSA hostkey for SSH version 1 connections.
<768-32768>	The length in bits of the generated key. The default is 1024 bits.

Default 1024 bits is the default key length. The DSA algorithm supports 1024 bits.

Mode Global Configuration

Examples To generate an RSA host key for SSH version 2 connections that is 2048 bits in length, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key generate hostkey rsa 2048
```

To generate a DSA host key, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key generate dsa
```

Related Commands
crypto key destroy hostkey
service ssh
show crypto key hostkey

crypto key generate userkey

This command generates public and private keys for an SSH user using either an RSA or DSA cryptography algorithm. To use public key authentication, copy the public key of the user onto the remote SSH server.

This command is not saved in the device configuration. However, the device saves the keys generated by this command in the non-volatile memory.

Syntax `crypto key generate userkey <username> {dsa|rsa|rsa1} [<768-32768>]`

Parameters	Description
`<username>`	Name of the user that the user key is generated for. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
`dsa`	Creates a DSA userkey. Both SSH version 1 and 2 connections can use a key created with this command.
`rsa`	Creates an RSA userkey for SSH version 2 connections.
`rsa1`	Creates an RSA userkey for SSH version 1 connections.
`<768-32768>`	The length in bits of the generated key. The DSA algorithm supports only 1024 bits. Default: 1024.

Mode Global Configuration

Examples To generate a 2048-bits RSA user key for SSH version 2 connections for the user `bob`, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key generate userkey bob rsa 2048
```

To generate a DSA user key for the user `lapo`, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key generate userkey lapo dsa
```

Related Commands **crypto key destroy userkey**
show crypto key userkey

crypto key pubkey-chain knownhosts

This command adds a public key of the specified SSH server to the known host database on your switch. The SSH client on your switch uses this public key to verify the remote SSH server.

The key is retrieved from the server. Before adding a key to this database, check that the key sent to you is correct.

If the server's key changes, or if your SSH client does not have the public key of the remote SSH server, then your SSH client will inform you that the public key of the server is unknown or altered.

The **no** variant of this command deletes the public key of the specified SSH server from the known host database on your device.

Syntax

```
crypto key pubkey-chain knownhosts [ip|ipv6] <hostname> [rsa|dsa|
   rsa1]

no crypto key pubkey-chain knownhosts <1-65535>
```

Parameter	Description
`ip`	Keyword used prior to specifying an IPv4 address
`ipv6`	Keyword used prior to specifying an IPv6 address
<hostname>	IPv4/IPv6 address or hostname of a remote server in the format `a.b.c.d` for an IPv4 address, or in the format `x:x::x:x` for an IPv6 address.
`rsa`	Specify the RSA public key of the server to be added to the known host database.
`dsa`	Specify the DSA public key of the server to be added to the known host database.
`rsa1`	Specify the SSHv1 public key of the server to be added to the know host database.
<1-65535>	Specify a key identifier when removing a key using the **no** parameter.

Default If no cryptography algorithm is specified, then **rsa** is used as the default cryptography algorithm.

Mode Privilege Exec

Usage This command adds a public key of the specified SSH server to the known host database on the switch. The key is retrieved from the server. The remote SSH server is verified by using this public key. The user is requested to check the key is correct before adding it to the database.

If the remote server's host key is changed, or if the device does not have the public key of the remote server, then SSH clients will inform the user that the public key of the server is altered or unknown.

Examples

To add the RSA host key of the remote SSH host IPv4 address `192.0.2.11` to the known host database, use the command:

```
awplus# crypto key pubkey-chain knownhosts 192.0.2.11
```

To delete the second entry in the known host database, use the command:

```
awplus# no crypto key pubkey-chain knownhosts 2
```

Validation Commands

show crypto key pubkey-chain knownhosts

crypto key pubkey-chain userkey

This command adds a public key for an SSH user on the SSH server. This allows the SSH server to support public key authentication for the SSH user. When configured, the SSH user can access the SSH server without providing a password from the remote host.

The **no** variant of this command removes a public key for the specified SSH user that has been added to the public key chain. When a SSH user's public key is removed, the SSH user can no longer login using public key authentication.

Syntax

```
crypto key pubkey-chain userkey <username> [<filename>]
no crypto key pubkey-chain userkey <username> <1-65535>
```

Parameters	Description
<username>	Name of the user that the SSH server associates the key with. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols. Default: no default
<filename>	Filename of a key saved in flash. Valid characters are any printable character. You can add a key as a hexadecimal string directly into the terminal if you do not specify a filename.
<1-65535>	The key ID number of the user's key. Specify the key ID to delete a key.

Mode Global Configuration

Usage You should import the public key file from the client node. The device can read the data from a file on the flash or user terminal.

Or you can add a key as text into the terminal. To add a key as text into the terminal, first enter the command **crypto key pubkey-chain userkey** <***username***>, and hit [Enter]. Enter the key as text. Note that the key you enter as text must be a valid SSH RSA key, not random ASCII text. Use [Ctrl]+D after entering it to save the text and re-enter the normal command line mode.

Note you can generate a valid SSH RSA key on the switch first using the **crypto key generate host rsa** command. View the SSH RSA key generated on the switch using the **show crypto hostkey rsa** command. Copy and paste the displayed SSH RSA key after entering the **crypto key pubkey-chain userkey** <***username***> command. Use [Ctrl]+D after entering it to save it.

Examples To generate a valid SSH RSA key on the switch and add the key, use the following commands:

```
awplus# configure terminal
awplus(config)# crypto key generate host rsa
awplus(config)# exit
awplus# show crypto key hostkey rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAr1s7SokW5aW2fcOw1TStpb9J
20bWluhnUC768EoWhyPW6FZ2t5360O5M29EpKBmGqlkQaz5V0mU9
IQe66+5YyD4UxOKSDtTI+7jtjDcoGWHb2u4sFwRpXwJZcgYrXW16
+6NvNbk+h+c/pqGDijj4SvfZZfeITzvvyZW4/I4pbN8=

awplus# configure terminal
awplus(config)# crypto key pubkey-chain userkey joe
Type CNTL/D to finish:
AAAAB3NzaC1yc2EAAAABIwAAAIEAr1s7SokW5aW2fcOw1TStpb9J
20bWluhnUC768EoWhyPW6FZ2t5360O5M29EpKBmGqlkQaz5V0mU9
IQe66+5YyD4UxOKSDtTI+7jtjDcoGWHb2u4sFwRpXwJZcgYrXW16
+6NvNbk+h+c/pqGDijj4SvfZZfeITzvvyZW4/I4pbN8=
control-D
awplus(config)#
```

To add a public key for the user `graydon` from the file `key.pub`, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key pubkey-chain userkey graydon key.pub
```

To add a public key for the user `tamara` from the terminal, use the commands:

```
awplus# configure terminal
awplus(config)# crypto key pubkey-chain userkey tamara
```

and enter the key. Use Ctrl+D to finish.

To remove the first key entry from the public key chain of the user `john`, use the commands:

```
awplus# configure terminal
awplus(config)# no crypto key pubkey-chain userkey john 1
```

Related Commands **show crypto key pubkey-chain userkey**

debug ssh client

This command enables the SSH client debugging facility. When enabled, any SSH, SCP and SFTP client sessions send diagnostic messages to the login terminal.

The **no** variant of this command disables the SSH client debugging facility. This stops the SSH client from generating diagnostic debugging message.

Syntax

```
debug ssh client [brief|full]
no debug ssh client
```

Parameter	Description
`brief`	Enables brief debug mode.
`full`	Enables full debug mode.

Default SSH client debugging is disabled by default.

Mode Privileged Exec and Global Configuration

Examples To start SSH client debugging, use the command:

```
awplus# debug ssh client
```

To start SSH client debugging with extended output, use the command:

```
awplus# debug ssh client full
```

To disable SSH client debugging, use the command:

```
awplus# no debug ssh client
```

Related Commands
debug ssh server
show ssh client
undebug ssh client

debug ssh server

This command enables the SSH server debugging facility. When enabled, the SSH server sends diagnostic messages to the system log. To display the debugging messages on the terminal, use the **terminal monitor** command.

The **no** variant of this command disables the SSH server debugging facility. This stops the SSH server from generating diagnostic debugging messages.

Syntax

```
debug ssh server [brief|full]
no debug ssh server
```

Parameter	Description
`brief`	Enables brief debug mode.
`full`	Enables full debug mode.

Default SSH server debugging is disabled by default.

Mode Privileged Exec and Global Configuration

Examples To start SSH server debugging, use the command:

```
awplus# debug ssh server
```

To start SSH server debugging with extended output, use the command:

```
awplus# debug ssh server full
```

To disable SSH server debugging, use the command:

```
awplus# no debug ssh server
```

Related Commands

debug ssh client
show ssh server
undebug ssh server

service ssh

This command enables the Secure Shell server on the device. Once enabled, connections coming from SSH clients are accepted.

SSH server needs a host key before it starts. If an SSHv2 host key does not exist, then this command fails. If SSHv1 is enabled but a host key for SSHv1 does not exist, then SSH service is unavailable for version 1.

The **no** variant of this command disables the Secure Shell server. When the Secure Shell server is disabled, connections from SSH, SCP, and SFTP clients are not accepted. This command does not affect existing SSH sessions. To terminate existing sessions, use the **clear ssh** command.

Syntax

```
service ssh [ip|ipv6]
no service ssh [ip|ipv6]
```

Default The Secure Shell server is disabled by default. Both IPv4 and IPv6 Secure Shell server are enabled when you issue **service ssh** without specifying the optional **ip** or **ipv6** parameters.

Mode Global Configuration

Examples To enable both the IPv4 and the IPv6 Secure Shell server, use the commands:

```
awplus# configure terminal
awplus(config)# service ssh
```

To enable the IPv4 Secure Shell server only, use the commands:

```
awplus# configure terminal
awplus(config)# service ssh ip
```

To enable the IPv6 Secure Shell server only, use the commands:

```
awplus# configure terminal
awplus(config)# service ssh ipv6
```

To disable both the IPv4 and the IPv6 Secure Shell server, use the commands:

```
awplus# configure terminal
awplus(config)# no service ssh
```

To disable the IPv4 Secure Shell server only, use the commands:

```
awplus# configure terminal
awplus(config)# no service ssh ip
```

To disable the IPv6 Secure Shell server only, use the commands:

```
awplus# configure terminal
awplus(config)# no service ssh ipv6
```

Related Commands

crypto key generate hostkey
show running-config ssh
show ssh server
ssh server allow-users
ssh server deny-users

show banner login

This command displays the banner message configured on the device. The banner message is displayed to the remote user before user authentication starts.

Syntax `show banner login`

Mode User Exec, Privileged Exec, Global Configuration, Interface Configuration, Line Configuration

Example To display the current login banner message, use the command:

```
awplus# show banner login
```

Related Commands banner login (SSH)

show crypto key hostkey

This command displays the SSH host keys generated by RSA and DSA algorithm.

A host key pair (public and private keys) is needed to enable SSH server. The private key remains on the device secretly. The public key is copied to SSH clients to identify the server

Syntax

```
show crypto key hostkey [dsa|rsa|rsa1]
```

Parameter	Description
dsa	Displays the DSA algorithm public key.
rsa	Displays the RSA algorithm public key for SSH version 2 connections.
rsa1	Displays the RSA algorithm public key for SSH version 1 connections.

Mode User Exec, Privileged Exec and Global Configuration

Examples To show the public keys generated on the device for SSH server, use the command:

```
awplus# show crypto key hostkey
```

To display the RSA public key of the SSH server, use the command:

```
awplus# show crypto key hostkey rsa
```

Output **Figure 78-1: Example output from the show crypto key hostkey command**

```
Type Bits   Fingerprint
-------------------------------------------------------------
rsa  2058   4e:7d:1d:00:75:79:c5:cb:c8:58:2e:f9:29:9c:1f:48
dsa  1024   fa:72:3d:78:35:14:cb:9a:1d:ca:1c:83:2c:7d:08:43
rsa1 1024   e2:1c:c8:8b:d8:6e:19:c8:f4:ec:00:a2:71:4e:85:8b
```

Table 78-1: Parameters in output of the show crypto key hostkey command

Parameter	Description
Type	Algorithm used to generate the key.
Bits	Length in bits of the key.
Fingerprint	Checksum value for the public key.

Related Commands crypto key destroy hostkey
crypto key generate hostkey

show crypto key pubkey-chain knownhosts

This command displays the list of public keys maintained in the known host database on the device.

Syntax `show crypto key pubkey-chain knownhosts [<1-65535>]`

Parameter	Description
<1-65535>	Key identifier for a specific key. Displays the public key of the entry if specified.

Default Display all keys.

Mode User Exec, Privileged Exec and Global Configuration

Examples To display public keys of known SSH servers, use the command:

```
awplus# show crypto key pubkey-chain knownhosts
```

To display the key data of the first entry in the known host data, use the command:

```
awplus# show crypto key pubkey-chain knownhosts 1
```

Output **Figure 78-2: Example output from the show crypto key public-chain knownhosts command**

```
No     Hostname         Type Fingerprint
-------------------------------------------------------------------------
1      172.16.23.1      rsa  c8:33:b1:fe:6f:d3:8c:81:4e:f7:2a:aa:a5:be:df:18
2      172.16.23.10     rsa  c4:79:86:65:ee:a0:1d:a5:6a:e8:fd:1d:d3:4e:37:bd
3      5ffe:1053:ac21:ff00:0101:bcdf:ffff:0001
                        rsa1 af:4e:b4:a2:26:24:6d:65:20:32:d9:6f:32:06:ba:57
```

Table 78-2: Parameters in the output of the show crypto key public-chain knownhosts command

Parameter	Description
`No`	Number ID of the key.
`Hostname`	Host name of the known SSH server.
`Type`	The algorithm used to generate the key.
`Fingerprint`	Checksum value for the public key.

Related Commands crypto key pubkey-chain knownhosts

show crypto key pubkey-chain userkey

This command displays the public keys registered with the SSH server for SSH users. These keys allow remote users to access the device using public key authentication. By using public key authentication, users can access the SSH server without providing password.

Syntax `show crypto key pubkey-chain userkey <username> [<1-65535>]`

Parameter	Description
<username>	User name of the remote SSH user whose keys you wish to display. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
<1-65535>	Key identifier for a specific key.

Default Display all keys.

Mode User Exec, Privileged Exec and Global Configuration

Example To display the public keys for the user `manager` that are registered with the SSH server, use the command:

```
awplus# show crypto key pubkey-chain userkey manager
```

Output **Figure 78-3: Example output from the show crypto key public-chain userkey command**

```
No Type Bits Fingerprint
--------------------------------------------------------------------
1  dsa  1024 2b:cc:df:a8:f8:2e:8f:a4:a5:4f:32:ea:67:29:78:fd
2  rsa  2048 6a:ba:22:84:c1:26:42:57:2c:d7:85:c8:06:32:49:0e
```

Table 78-3: Parameters in the output of the show crypto key userkey command

Parameter	Description
`No`	Number ID of the key.
`Type`	The algorithm used to generate the key.
`Bits`	Length in bits of the key.
`Fingerprint`	Checksum value for the key.

Related Commands crypto key pubkey-chain userkey

show crypto key userkey

This command displays the public keys created on this device for the specified SSH user.

Syntax `show crypto key userkey <username> [dsa|rsa|rsa1]`

Parameter	Description
<username>	User name of the local SSH user whose keys you wish to display. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen and full stop symbols.
`dsa`	Displays the DSA public key.
`rsa`	Displays the RSA public key used for SSH version 2 connections.
`rsa1`	Displays the RSA key used for SSH version 1 connections.

Mode User Exec, Privileged Exec and Global Configuration

Examples To show the public key generated for the user, use the command:

```
awplus# show crypto key userkey manager
```

To store the RSA public key generated for the user manager to the file "user.pub", use the command:

```
awplus# show crypto key userkey manager rsa > manager-rsa.pub
```

Output **Figure 78-4: Example output from the show crypto key userkey command**

```
Type Bits   Fingerprint
-------------------------------------------------------------
rsa  2048   e8:d6:1b:c0:f4:b6:e6:7d:02:2e:a9:d4:a1:ca:3b:11
rsa1 1024   12:25:60:95:64:08:8e:a1:8c:3c:45:1b:44:b9:33:9b
```

Table 78-4: Parameters in the output of the show crypto key userkey command

Parameter	Description
`Type`	The algorithm used to generate the key.
`Bits`	Length in bits of the key.
`Fingerprint`	Checksum value for the key.

Related Commands crypto key generate userkey

show running-config ssh

This command displays the current running configuration of Secure Shell (SSH).

Syntax `show running-config ssh`

Mode Privileged Exec and Global Configuration

Example To display the current configuration of SSH, use the command:

```
awplus# show running-config ssh
```

Output **Figure 78-5: Example output from the show running-config ssh command**

```
!
ssh server session-timeout 600
ssh server login-timeout 30
ssh server allow-users manager 192.168.1.*
ssh server allow-users john
ssh server deny-user john*.a-company.com
ssh server
```

Table 78-5: Parameters in the output of the show running-config ssh command

Parameter	Description
`ssh server`	SSH server is enabled.
`ssh server v2`	SSH server is enabled and only support SSHv2.
`ssh server <port>`	SSH server is enabled and listening on the specified TCP port.
`no ssh server scp`	SCP service is disabled.
`no ssh server sftp`	SFTP service is disabled.
`ssh server session-timeout`	Configure the server session timeout.
`ssh server login-timeout`	Configure the server login timeout.
`ssh server max-startups`	Configure the maximum number of concurrent sessions waiting authentication.
`no ssh server authentication password`	Password authentication is disabled.
`no ssh server authentication publickey`	Public key authentication is disabled.
`ssh server allow-users`	Add the user (and hostname) to the allow list.
`ssh server deny-users`	Add the user (and hostname) to the deny list.

Related Commands **service ssh**
show ssh server

show ssh

This command displays the active SSH sessions on the device, both incoming and outgoing.

Syntax `show ssh`

Mode User Exec, Privileged Exec and Global Configuration

Example To display the current SSH sessions on the device, use the command:

```
awplus# show ssh
```

Output **Figure 78-6: Example output from the show ssh command**

```
Secure Shell Sessions:
ID  Type Mode   Peer Host    Username   State       Filename
---------------------------------------------------------------
414 ssh  server 172.16.23.1  root       open
456 ssh  client 172.16.23.10 manager    user-auth
459 scp  client 172.16.23.12 root       download    550dev_.awd
463 ssh  client 5ffe:33fe:5632:ffbb:bc35:ddee:0101:ac51
                             manager    user-auth
```

Table 78-6: Parameters in the output of the show ssh command

Parameter	Description	
ID	Unique identifier for each SSH session.	
Type	Session type; either SSH, SCP, or SFTP.	
Mode	Whether the device is acting as an SSH client (client) or SSH server (server) for the specified session.	
Peer Host	The hostname or IP address of the remote server or client.	
Username	Login user name of the server.	
State	The current state of the SSH session. One of:	
	connecting	The device is looking for a remote server.
	connected	The device is connected to the remote server.
	accepted	The device has accepted a new session.
	host-auth	host-to-host authentication is in progress.
	user-auth	User authentication is in progress.
	authenticated	User authentication is complete.
	open	The session is in progress.
	download	The user is downloading a file from the device.
	upload	The user is uploading a file from the device.
	closing	The user is terminating the session.
	closed	The session is closed.
Filename	Local filename of the file that the user is downloading or uploading.	

Related Commands clear ssh

show ssh client

This command displays the current configuration of the Secure Shell client.

Syntax `show ssh client`

Mode User Exec, Privileged Exec and Global Configuration

Example To display the current configuration for SSH clients on the login shell, use the command:

```
awplus# show ssh client
```

Output **Figure 78-7: Example output from the show ssh client command**

```
Secure Shell Client Configuration
-----------------------------------------------------------------
Port                                : 22
Version                             : 2,1
Connect Timeout                     : 30 seconds
Session Timeout                     : 0 (off)
Debug                               : NONE
```

Table 78-7: Parameters in the output of the show ssh client command

Parameter	Description
`Port`	SSH server TCP port where the SSH client connects to. The default is port 22.
`Version`	SSH server version; either “1”, “2” or “2,1”.
`Connect Timeout`	Time in seconds that the SSH client waits for an SSH session to establish. If the value is 0, the connection is terminated when it reaches the TCP timeout.
`Debug`	Whether debugging is active on the client.

Related Commands **show ssh server**

show ssh server

This command displays the current configuration of the Secure Shell server.

Note that changes to the SSH configuration affects only new SSH sessions coming from remote hosts, and does not affect existing sessions.

Syntax `show ssh server`

Mode User Exec, Privileged Exec and Global Configuration

Example To display the current configuration of the Secure Shell server, use the command:

```
awplus# show ssh server
```

Output **Figure 78-8: Example output from the show ssh server command**

```
Secure Shell Server Configuration
-----------------------------------------------------------------
SSH Server                          : Enabled
Port                                : 22
Version                             : 2
Services                            : scp, sftp
User Authentication                 : publickey, password
Idle Timeout                        : 60 seconds
Maximum Startups                    : 10
Debug                               : NONE
```

Table 78-8: Parameters in the output of the show ssh server command

Parameter	Description
`SSH Server`	Whether the Secure Shell server is enabled or disabled.
`Port`	TCP port where the Secure Shell server listens for connections. The default is port 22.
`Version`	SSH server version; either “1”, “2” or “2,1”.
`Services`	List of the available Secure Shell service; one or more of SHELL, SCP or SFTP.
`Authentication`	List of available authentication methods.
`Login Timeout`	Time (in seconds) that the SSH server will wait the SSH session to establish. If the value is 0, the client login will be terminated when TCP timeout reaches.
`Idle Timeout`	Time (in seconds) that the SSH server will wait to receive data from the SSH client. The server disconnects if this timer limit is reached. If set at 0, the idle timer remains off.
`Maximum Startups`	The maximum number of concurrent connections that are waiting authentication. The default is 10.
`Debug`	Whether debugging is active on the server.

Related Commands **show ssh**
show ssh client

show ssh server allow-users

This command displays the user entries in the allow list of the SSH server.

Syntax `show ssh server allow-users`

Mode User Exec, Privileged Exec and Global Configuration

Example To display the user entries in the allow list of the SSH server, use the command:

```
awplus# show ssh server allow-users
```

Output **Figure 78-9: Example output from the show ssh server allow-users command**

```
Username          Remote Hostname (pattern)
----------------- --------------------------------
awplus            192.168.*
john
manager           *.alliedtelesis.com
```

Table 78-9: Parameters in the output of the show ssh server allow-users command

Parameter	Description
`Username`	User name that is allowed to access the SSH server.
`Remote Hostname (pattern)`	IP address or hostname pattern of the remote client. The user is allowed requests from a host that matches this pattern. If no hostname is specified, the user is allowed from all hosts.

Related Commands **ssh server allow-users**
ssh server deny-users

show ssh server deny-users

This command displays the user entries in the deny list of the SSH server. The user in the deny list is rejected to access the SSH server. If a user is not included in the access list of the SSH server, the user is also rejected.

Syntax `show ssh server deny-users`

Mode User Exec, Privileged Exec and Global Configuration

Example To display the user entries in the deny list of the SSH server, use the command:

```
awplus# show ssh server deny-users
```

Output **Figure 78-10: Example output from the show ssh server deny-user command**

```
Username          Remote Hostname (pattern)
----------------- ---------------------------------
john              *.b-company.com
manager           192.168.2.*
```

Table 78-10: Parameters in the output of the show ssh server deny-user command

Parameter	Description
`Username`	The user that this rule applies to.
`Remote Hostname (pattern)`	IP address or hostname pattern of the remote client. The user is denied requests from a host that matches this pattern. If no hostname is specified, the user is denied from all hosts.

Related Commands ssh server allow-users
ssh server deny-users

ssh

This command initiates a Secure Shell connection to a remote SSH server.

If the server requests a password for the user login, the user needs to type in the correct password on “Password:” prompt.

SSH client identifies the remote SSH server by its public key registered on the client device. If the server identification is changed, server verification fails. If the public key of the server has been changed, the public key of the server must be explicitly added to the known host database.

Note Note that any hostname specified with ssh cannot begin with a hyphen (-) character.

Syntax

```
ssh [ip|ipv6][{[user <username>]|[port <1-65535>]|[version {1|2}]}]
   <hostname> [<line>]
```

Parameter	Description	
`ip`	Specify IPv4 SSH.	
`ipv6`	Specify IPv6 SSH.	
`user`	Login user. If user is specified, the username is used for login to the remote SSH server when user authentication is required. Otherwise the current user name is used.	
	<username>	User name to login on the remote server.
`port`	SSH server port. If port is specified, the SSH client connects to the remote SSH server with the specified TCP port. Other- wise, the client port configured by “ssh client” command or the default TCP port (22) is used.	
	<1-65535>	TCP port.
`version`	SSH client version. If version is specified, the SSH client supports only the specified SSH version. By default, SSH client uses SSHv2 first. If the server does not support SSHv2, it will try SSHv1. The default version can be configured by “ssh client” command.	
	`1`	Use SSH version 1.
	`2`	Use SSH version 2.

Parameter(cont.)	Description(cont.)	
<hostname>	IPv4/IPv6 address or hostname of a remote server in the format `a.b.c.d` for an IPv4 address, or in the format `x:x::x:x` for an IPv6 address corresponding to the ip or ipv6 optional keywords used. Note that any hostname specified with ssh cannot begin with a hyphen (-) character.	
	<line>	Command to execute on the remote server. If a command is specified, the command is executed on the remote SSH server and the session is disconnected when the remote command finishes.

Mode User Exec and Privileged Exec

Examples To login to the remote SSH server at 192.0.2.5, use the command:

```
awplus# ssh ip 192.0.2.5
```

To login to the remote SSH server at 192.0.2.5 as user **manager**, use the command:

```
awplus# ssh ip user manager 192.0.2.5
```

To login to the remote SSH server at 192.0.2.5 that is listening TCP port 2000, use the command:

```
awplus# ssh port 2000 192.0.2.5
```

To login to the remote SSH server with example_host using IPv6 session, use the command:

```
awplus# ssh ipv6 example_host
```

To run the **cmd** command on the remote SSH server at 192.0.2.5, use the command:

```
awplus# ssh ip 192.0.2.5 cmd
```

Related Commands
crypto key generate userkey
crypto key pubkey-chain knownhosts
debug ssh client
ssh client

ssh client

This command modifies the default configuration parameters of the Secure Shell (SSH) client. The configuration is used for any SSH client on the device to connect to remote SSH servers. Any parameters specified on SSH client explicitly override the default configuration parameters.

The change affects the current user shell only. When the user exits the login session, the configuration does not persist. This command does not affect existing SSH sessions.

The **no** variant of this command resets configuration parameters of the Secure Shell (SSH) client changed by the **ssh client** command, and restores the defaults.

This command does not affect the existing SSH sessions.

Syntax

```
ssh client {port <1-65535>|version {1|2}|session-timeout <0-3600>|
    connect-timeout <1-600>}

no ssh client {port|version|session-timeout|connect-timeout}
```

Parameter	Description	
`port`	The default TCP port of the remote SSH server. If an SSH client specifies an explicit port of the server, it overrides the default TCP port. Default: **22**	
	<1-65535>	TCP port number.
`version`	The SSH version used by the client for SSH sessions. The SSH client supports both version 2 and version 1 Default: **version 2** **Note**: SSH version 2 is the default SSH version. SSH client supports SSH version 1 if SSH version 2 is not configured using a **ssh version** command.	
	`1`	SSH clients on the device supports SSH version 1 only.
	`2`	SSH clients on the device supports SSH version 2 only
`session-timeout`	The global session timeout for SSH sessions. If the session timer lapses since the last time an SSH client received data from the remote server, the session is terminated. If the value is 0, then the client does not terminate the session. Instead, the connection is terminated when it reaches the TCP timeout. Default: **0** (session timer remains off)	
	<0-3600>	Timeout in seconds.
`connect-timeout`	The maximum time period that an SSH session can take to become established. The SSH client terminates the SSH session if this timeout expires and the session is still not established. Default: **30**	
	<1-600>	Timeout in seconds.

Mode Privileged Exec

Examples

To configure the default TCP port for SSH clients to 2200, and the session timer to 10 minutes, use the command:

```
awplus# ssh client port 2200 session-timeout 600
```

To configure the connect timeout of SSH client to 10 seconds, use the command:

```
awplus# ssh client connect-timeout 10
```

To restore the connect timeout to its default, use the command:

```
awplus# no ssh client connect-timeout
```

Related Commands

show ssh client
ssh

ssh server

This command modifies the configuration of the SSH server. Changing these parameters affects new SSH sessions connecting to the device.

The **no** variant of this command restores the configuration of a specified parameter to its default. The change affects the SSH server immediately if the server is running. Otherwise, the configuration is used when the server starts.

To enable the SSH server, use the **service ssh** command.

Syntax

```
ssh server {[v1v2|v2only]|<1-65535>}
ssh server {[session-timeout <0-3600>] [login-timeout <1-600>]
    [max-startups <1-128>]}
no ssh server {[session-timeout] [login-timeout] [max-startups]}
```

Parameter	Description
v1v2	Supports both SSHv2 and SSHv1client connections. Default: **v1v2**
v2only	Supports SSHv2 client connections only.
<1-65535>	The TCP port number that the server listens to for incoming SSH sessions. Default: **22**
session-timeout	There is a maximum time period that the server waits before deciding that a session is inactive and should be terminated. The server considers the session inactive when it has not received any data from the client, and when the client does not respond to keep alive messages. Default: **0** (session timer remains off).
	<0-3600> Timeout in seconds.
login-timeout	The maximum time period the server waits before disconnecting an unauthenticated client. Default: **60**
	<1-600> Timeout in seconds.
max-startups	The maximum number of concurrent unauthenticated connections the server accepts. When the number of SSH connections awaiting authentication reaches the limit, the server drops any additional connections until authentication succeeds or the login timer expires for a connection. Default: **10**
	<1-128> Number of sessions.

Mode Global Configuration

Examples To configure the session timer of SSH server to 10 minutes (600 seconds), use the commands:

```
awplus# configure terminal
awplus(config)# ssh server login-timeout 600
```

To configure the login timeout of SSH server to 30 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server login-timeout 30
```

To limit the number of SSH client connections waiting authentication from SSH server to 3, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server max-startups
```

To set max-startups parameters of SSH server to the default configuration, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server max-startups
```

To support the Secure Shell server with TCP port 2200, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server 2200
```

To force the Secure Shell server to support SSHv2 only, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server v2only
```

To support both SSHv2 and SSHv1, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server v1v2
```

Related Commands **show ssh server**
ssh client

ssh server allow-users

This command adds a username pattern to the allow list of the SSH server. If the user of an incoming SSH session matches the pattern, the session is accepted.

When there are no registered users in the server's database of allowed users, the SSH server does not accept SSH sessions even when enabled.

SSH server also maintains the deny list. The server checks the user in the deny list first. If a user is listed in the deny list, then the user access is denied even if the user is listed in the allow list.

The **no** variant of this command deletes a username pattern from the allow list of the SSH server. To delete an entry from the allow list, the username and hostname pattern should match exactly with the existing entry.

Syntax

```
ssh server allow-users <username-pattern> [<hostname-pattern>]
no ssh server allow-users <username-pattern> [<hostname-pattern>]
```

Parameter	Description
<username-pattern>	The username pattern that users can match to. An asterisk acts as a wildcard character that matches any string of characters.
<*hostname-pattern*>	The host name pattern that hosts can match to. If specified, the server allows the user to connect only from hosts matching the pattern. An asterisk acts as a wildcard character that matches any string of characters.

Mode Global Configuration

Examples To allow the user `john` to create an SSH session from any host, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server allow-users john
```

To allow the user `john` to create an SSH session from a range of IP address (from 192.168.1.1 to 192.168.1.255), use the commands:

```
awplus# configure terminal
awplus(config)# ssh server allow-users john 192.168.1.*
```

To allow the user `john` to create a SSH session from `a-company.com` domain, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server allow-users john *.a-company.com
```

To delete the existing user entry `john 192.168.1.*` in the allow list, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server allow-users john 192.168.1.*
```

Related Commands

show running-config ssh
show ssh server allow-users
ssh server deny-users

ssh server authentication

This command enables RSA public-key or password user authentication for SSH Server. Apply the **password** keyword with the **ssh server authentication** command to enable password authentication for users. Apply the **publickey** keyword with the **ssh server authentication** command to enable RSA public-key authentication for users.

Use the **no** variant of this command to disable RSA public-key or password user authentication for SSH Server. Apply the **password** keyword with the **no ssh authentication** command to disable password authentication for users. Apply the required **publickey** keyword with the **no ssh authentication** command to disable RSA public-key authentication for users.

Syntax

```
ssh server authentication {password|publickey}
no ssh server authentication {password|publickey}
```

Parameter	Description
`password`	Specifies user password authentication for SSH server.
`publickey`	Specifies user publickey authentication for SSH server.

Default Both RSA public-key authentication and password authentication are enabled by default.

Mode Global Configuration

Usage For password authentication to authenticate a user, password authentication for a user must be registered in the local user database or on an external RADIUS server, before using the **ssh server authentication password** command.

For RSA public-key authentication to authenticate a user, a public key must be added for the user, before using the **ssh server authentication publickey** command.

Examples To enable `password` authentication for users connecting through SSH, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server authentication password
```

To enable `publickey` authentication for users connecting through SSH, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server authentication publickey
```

To disable `password` authentication for users connecting through SSH, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server authentication password
```

To disable `publickey` authentication for users connecting through SSH, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server authentication publickey
```

Related Commands

crypto key pubkey-chain userkey
service ssh
show ssh server

ssh server deny-users

This command adds a username pattern to the deny list of the SSH server. If the user of an incoming SSH session matches the pattern, the session is rejected.

SSH server also maintains the allow list. The server checks the user in the deny list first. If a user is listed in the deny list, then the user access is denied even if the user is listed in the allow list.

If a hostname pattern is specified, the user is denied from the hosts matching the pattern.

The **no** variant of this command deletes a username pattern from the deny list of the SSH server. To delete an entry from the deny list, the username and hostname pattern should match exactly with the existing entry.

Syntax

```
ssh server deny-users <username-pattern> [<hostname-pattern>]
no ssh server deny-users <username-pattern> [<hostname-pattern>]
```

Parameter	Description
<username-pattern>	The username pattern that users can match to. The username must begin with a letter. Valid characters are all numbers, letters, and the underscore, hyphen, full stop and asterisk symbols. An asterisk acts as a wildcard character that matches any string of characters.
<hostname-pattern>	The host name pattern that hosts can match to. If specified, the server denies the user only when they connect from hosts matching the pattern. An asterisk acts as a wildcard character that matches any string of characters.

Mode Global Configuration

Examples To deny the user `john` to access SSH login from any host, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server deny-users john
```

To deny the user john to access SSH login from a range of IP address (from 192.168.2.1 to 192.168.2.255), use the commands:

```
awplus# configure terminal
awplus(config)# ssh server deny-users john 192.168.2.*
```

To deny the user `john` to access SSH login from `b-company.com` domain, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server deny-users john*.b-company.com
```

To delete the existing user entry `john 192.168.2.*` in the deny list, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server deny-users john 192.168.2.*
```

Related Commands
show running-config ssh
show ssh server deny-users
ssh server allow-users

ssh server resolve-host

This command enables resolving an IP address from a host name using a DNS server for client host authentication.

The **no** variant of this command disables this feature.

Syntax

```
ssh server resolve-hosts
no ssh server resolve-hosts
```

Default This feature is disabled by default.

Mode Global Configuration

Usage Your device has a DNS Client that is enabled automatically when you add a DNS server to your device. To add a DNS server to the list of servers that the device sends DNS queries to use the **ip name-server** command on page 29.31.

For information about configuring DNS see **"Domain Name System (DNS)" on page 28.8**.

Example To resolve a host name using a DNS server, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server resolve-hosts
```

Related Commands
ip name-server
show ssh server
ssh server allow-users
ssh server deny-users

ssh server scp

This command enables the Secure Copy (SCP) service on the SSH server. Once enabled, the server accepts SCP requests from remote clients.

You must enable the SSH server as well as this service before the device accepts SCP connections. The SCP service is enabled by default as soon as the SSH server is enabled.

The **no** variant of this command disables the SCP service on the SSH server. Once disabled, SCP requests from remote clients are rejected.

Syntax

```
ssh server scp
no ssh server scp
```

Mode Global Configuration

Examples To enable the SCP service, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server scp
```

To disable the SCP service, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server scp
```

Related Commands **show running-config ssh**
show ssh server

ssh server sftp

This command enables the Secure FTP (SFTP) service on the SSH server. Once enabled, the server accepts SFTP requests from remote clients.

You must enable the SSH server as well as this service before the device accepts SFTP connections. The SFTP service is enabled by default as soon as the SSH server is enabled. If the SSH server is disabled, SFTP service is unavailable.

The **no** variant of this command disables SFTP service on the SSH server. Once disabled, SFTP requests from remote clients are rejected.

Syntax

```
ssh server sftp
no ssh server sftp
```

Mode Global Configuration

Examples To enable the SFTP service, use the commands:

```
awplus# configure terminal
awplus(config)# ssh server sftp
```

To disable the SFTP service, use the commands:

```
awplus# configure terminal
awplus(config)# no ssh server sftp
```

Related Commands

show running-config ssh
show ssh server

undebug ssh client

This command applies the functionality of the **no debug ssh client** command.

undebug ssh server

This command applies the functionality of the **no debug ssh server** command.

Chapter 79: DHCP Snooping Introduction and Configuration

Introduction 79.2
DHCP Snooping 79.2
- DHCP Snooping Database 79.3
- DHCP Relay Agent Option 82 ..... 79.4
- Traffic Filtering with DHCP Snooping 79.6
- ARP Security 79.8
- MAC Address Verification 79.8
- DHCP Snooping Violations 79.8

Interactions with Other Features 79.9
Configuration 79.10
- Configure DHCP Snooping 79.10
- Disabling DHCP Snooping 79.16
- Related Features 79.16

Introduction

This chapter provides information about DHCP snooping, support for it on this switch, and how to configure it.

For detailed descriptions of the commands used to configure DHCP snooping, see **Chapter 80, DHCP Snooping Commands**; for related ACL commands, see **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands**.

For information about Dynamic Host Configuration protocol and how to configure it, see **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction** and **Chapter 90, Dynamic Host Configuration Protocol (DHCP) Commands**.

DHCP Snooping

DHCP snooping provides an extra layer of security on the switch via dynamic IP source filtering. DHCP snooping filters out traffic received from unknown, or 'untrusted' ports, and builds and maintains a DHCP snooping database.

Dynamic Host Configuration Protocol (DHCP) dynamically assigns IP addresses to client devices. The use of dynamically assigned addresses requires traceability, so that a service provider can determine which clients own a particular IP address at a certain time.

With DHCP snooping, IP sources are dynamically verified, and filtered accordingly. IP packets that are not sourced from recognized IP addresses can be filtered out. This ensures the required traceability.

With DHCP snooping, an administrator can control port-to-IP connectivity by:

- permitting port access to specified IP addresses only
- permitting port access to DHCP issued IP addresses only
- dictating the number of IP clients on any given port
- passing location information about an IP client to the DHCP server
- permitting only known IP clients to ARP

Ports on the switch are classified as either trusted or untrusted:

- Trusted ports receive only messages from within your network.
- Untrusted ports receive messages from outside your network.

DHCP snooping blocks unauthorized IP traffic from untrusted ports, and prevents it from entering the trusted network. It validates DHCP client packets from untrusted ports and forwards them to trusted ports in the VLAN.

On this switch, DHCP snooping is disabled by default, and can be enabled on per-VLAN basis to operate over switch ports and over static and dynamic (LACP) link aggregators (channel groups).

DHCP Snooping Database

When you enable DHCP snooping, the switch intercepts all DHCP packets it receives, and sends them to the Central Processing Unit (CPU), where they are verified. The DHCP snooping database stores and maintains this information. The database contains entries for:

- current IP address leases dynamically allocated by a DHCP server
- static or dynamic entries added from the command line—typically used to add a DHCP snooping entry for a client that has a preconfigured IP address on an untrusted port

Database backup

The switch periodically saves the dynamic entries in the DHCP snooping database to a hidden file (**.dhcp.dsn.gz**) in Non-Volatile Storage (NVS), or can be configured to save them to a USB storage device.

If such a database file exists, it is loaded when the switch starts up with DHCP snooping enabled, or when DHCP snooping is subsequently enabled.

Lease entries

Each entry in the database corresponds to a DHCP IP address lease.

For dynamic entries added automatically by DHCP snooping, each entry contains the following information:

- the IP address that was allocated to that client
- the MAC address of the client device
- the time until expiry
- the VLAN to which the client is attached
- the port to which the client is attached
- the IP address of the DHCP server

For static entries added from the command line, each entry contains the following subset of information:

- the IP address allocated to the client
- the MAC address of the client device (optional)
- the VLAN to which the client is attached
- the port to which the client is attached

Each entry also shows its source: Dynamic or Static.

On this switch, the maximum number of lease entries that can be stored in the DHCP snooping database for each port can be configured—the default is 1.

Expired entries

For dynamic entries, the switch receives expiry information with the client lease information in DHCP packets. Entries expire when the time left to expiry is 0 seconds. Expired entries are automatically deleted from the database. Static entries have no expiry information, and are not checked. All dynamic entries in the database are written to the backup file. Whenever DHCP snooping is enabled, the DHCP snooping database is repopulated from the backup file and any static entries in the start-up configuration file. Any entries present in the backup file that have expired are ignored.

DHCP Relay Agent Option 82

If the switch is at the edge of the network, it can be configured to insert DHCP Relay Agent Option 82 information into client-originated BOOTP/DHCP packets that it is forwarding to a DHCP server. The switch also removes DHCP Relay Agent Option 82 information from BOOTP reply packets destined for an untrusted port if the DHCP client hardware is directly attached to a port on the switch.

DHCP servers that are configured to recognize DHCP Relay Agent Option 82 may use the information to implement IP address or other parameter assignment policies, based on the network location of the client device.

When DHCP Relay Agent Option 82 information for DHCP snooping is enabled, the switch inserts DHCP Relay Agent Option 82 information into BOOTP request packets received from an untrusted port. The switch inserts the following DHCP Relay Agent Option 82 information:

- Remote ID: this identifies the host. By default, this is the MAC address of the switch (sub-option1).
- Circuit ID: this specifies the switch port and VLAN ID that the client-originated DHCP packet was received on (sub-option2). By default, this is the VLAN ID and the Ifindex (interface number).
- Subscriber ID (optional): this is a string of up to 50 characters that differentiates or groups client ports on the switch (sub-option 6).

Support on this switch

This switch inserts DHCP Relay Agent Option 82 (agent option) information into DHCP packets received through untrusted ports, and removes it from DHCP packets transmitted through untrusted ports. This is enabled by default, and can be disabled if required.

You can specify values for the Remote ID and Circuit ID sub-options of the DHCP Relay Agent Option 82 field. The Remote ID can be specified as an alphanumeric (ASCII) string, 1 to 63 characters in length. The Circuit ID can be specified as the VLAN ID and port number.

Subscriber IDs can be configured for ports, and if they have been configured, they are inserted in DHCP packets as part of the DHCP Relay Agent Option 82 information.

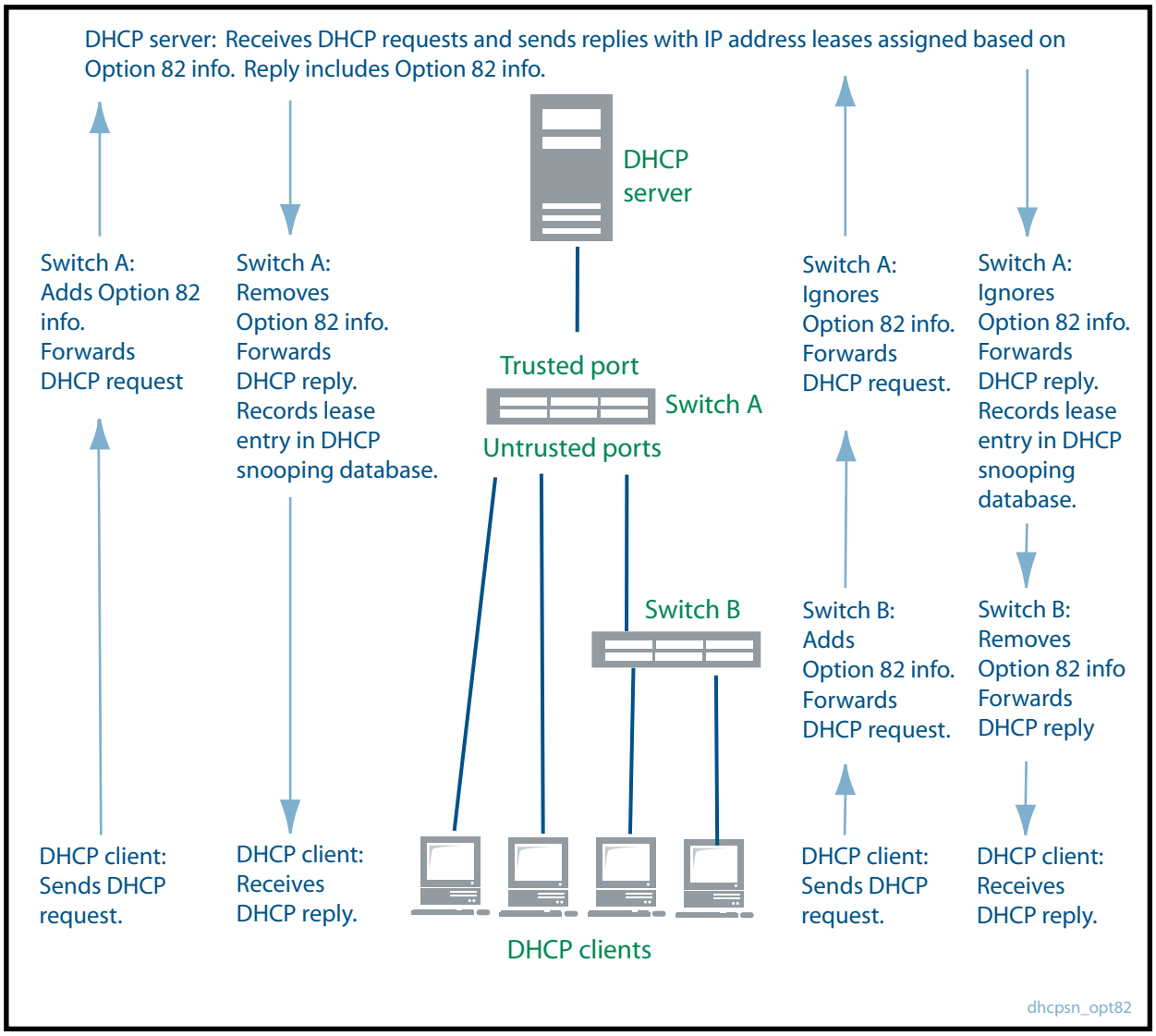
Regardless of whether DHCP Relay Agent Option 82 is enabled for DHCP snooping, if the switch receives a BOOTP/DHCP request packet on a trusted port, and the packet contains DHCP Relay Agent Option 82 information, it does not update the DHCP Relay Agent Option 82 information for the receiver port. By default, if it receives a DHCP request packet containing DHCP Relay Agent Option 82 information on an untrusted port, it drops the packet. However, if the switch is connected via untrusted ports to edge switches that insert DHCP Relay Agent Option 82 information into DHCP packets, you may need to allow these DHCP packets through the untrusted ports—the switch can be configured to forward these packets.

Note that the DHCP Relay Agent Option 82 agent information inserted by the DHCP snooping differs from the information added by DHCP Relay (see **"DHCP Relay Agent Introduction" on page 89.9**). The switch cannot be configured to use both the DHCP relay agent option and DHCP snooping.

Operation **Figure 79-1** shows DHCP packet flow between DHCP clients and server, where:

- Switch A has DHCP snooping enabled. The DHCP server is connected to a trusted port on Switch A; DHCP clients and Switch B are connected to untrusted ports.
- Switch A is configured to add and remove DHCP Relay Agent Option 82 information (**ip dhcp snooping agent-option** command on page 80.9).
- Switch A is configured to forward DHCP packets that already contain DHCP Relay Agent Option 82 information without changing it (**ip dhcp snooping agent-option allow-untrusted** command on page 80.10).
- Switch B is Layer 2 switching traffic from downstream DHCP clients, and adds and removes DHCP Relay Agent Option 82 information.

Figure 79-1: DHCP packet flow with DHCP snooping and DHCP Relay Agent Option 82 (agent option)



For more information about DHCP Relay Agent Option 82, see RFC 3046, DHCP Relay Agent Information Option.

Traffic Filtering with DHCP Snooping

DHCP filtering prevents IP addresses from being falsified or 'spoofed'. This guarantees that users cannot avoid detection by spoofing IP addresses that are not actually allocated to them. With DHCP filtering, the switch permits packets to enter over a specific port if their source IP address is currently allocated to a client connected to that port.

Support on this switch

On this switch, Access Control Lists (ACLs) based on DHCP snooping can be used with access groups to filter IP packets. For instance, IP traffic on untrusted ports can be limited to packets matching valid DHCP lease information stored in the DHCP snooping database. Quality of Service (QoS) configuration can also be applied to these ACLs.

The DHCP snooping feature is enabled or disabled per VLAN, and several of the related configuration settings are applied per port. If there are multiple VLANs on a port, all the VLANs will be subject to the same per-port settings.

Operation

Table 79-1 on page 79.7 shows the filtering that is applied by DHCP snooping on a switch with the following DHCP filtering configuration for untrusted ports:

- DHCP snooping is enabled on all VLANs (**service dhcp-snooping** command on page 80.23, **ip dhcp snooping** command on page 80.8)
- ARP security (**arp security** command on page 80.2) is enabled on all VLANs
- MAC address verification is enabled on the switch (**ip dhcp snooping verify mac-address** command on page 80.20; enabled by default), and all DHCP clients are directly connected to the switch.
- Access Control Lists allow IP packets that match the source IP address and MAC address of a valid lease entry in the DHCP snooping database, and deny other IP packets (**access-list** commands in **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands**).
- DHCP requests containing DHCP Relay Agent Option 82 info are not allowed (**ip dhcp snooping agent-option allow-untrusted** command on page 80.10 this is disabled by default).
- Log messages and SNMP notifications are enabled for DHCP snooping and ARP security violations (**ip dhcp snooping violation** command on page 80.21, **arp security violation** command on page 80.3, **snmp-server enable trap** command on page 94.18).

Table 79-1: DHCP filtering on the switch

When the switch ...	and ...	then the switch ...
DHCP packets		
Receives a DHCP BOOTP packet on a trusted port		Forwards the DHCP packet.
	The packet contains a valid IP address lease for a client, and the maximum number of leases for the client port has not been reached.	Adds or updates a lease entry in the DHCP snooping database.
	The maximum number of leases for the client port has been reached.	Drops the DHCP packet, generates a log message for the violation, generates an SNMP notification (trap), and does not add a lease entry to the database.
A lease entry in the DHCP snooping database expires		Removes the expired entry from the database.
Receives a DHCP BOOTP request packet on an untrusted port	The source MAC address and client hardware address do not match.	Drops the packet, generates a log message for the violation, and sends an SNMP notification (trap).
Receives a DHCP BOOTP request packet on an untrusted port	The packet contains DHCP Relay Agent Option 82 info.	Drops the DHCP packet, generates a log message for the violation, and sends an SNMP notification (trap).
Receives a DHCP BOOTP reply packet on an untrusted port		Drops the DHCP packet, generates a log message for the violation, and sends an SNMP notification (trap).
IP packets		
Receives an IP packet on a trusted port		Forwards the IP packet.
Receives an IP packet on an untrusted port	Its source MAC address, IP address, and receiving port match a valid lease entry in the DHCP snooping database.	Forwards the IP packet.
Receives an IP packet on an untrusted port	Its source MAC address, IP address, and receiving port do not match a valid lease entry in the DHCP snooping database.	Drops the packet. Does not generate a log message or an SNMP notification.
ARP packets		
Receives an ARP request on a trusted port		Forwards the ARP packet.
Receives an ARP request on an untrusted port	Its source MAC address, IP address, and receiving port match a valid entry in the DHCP snooping database	Forwards the ARP packet.
Receives an ARP request on an untrusted port	Its source MAC address, IP address, and receiving port do not match an entry in the DHCP snooping database	Drops the packet, generates a log message for the violation, and sends an SNMP notification (trap).

ARP Security

ARP security prevents ARP spoofing. ARP spoofing occurs when devices send fake, or 'spoofed', ARP messages to an Ethernet LAN. This makes it possible for an unauthorized host to claim to be an authorized host. The unauthorized host can then intercept traffic intended for the authorized host, and can access the wider network.

Spoofed ARP messages contain the IP address of an authorized host, with a MAC address which does not match the real MAC address of the host. When ARP security is enabled for DHCP snooping, the switch checks ARP packets sourced from untrusted ports against the entries in the DHCP snooping binding database. If it finds a matching entry, it forwards the ARP packet as normal. If it does not find a matching entry, it drops the ARP packet. This ensures that only trusted clients (with a recognized IP address and MAC address) can generate ARP packets into the network. ARP security is not applied to packets received on trusted ports.

ARP security is disabled by default, and can be enabled on VLANs to ensure that on untrusted ports, only trusted clients (with a recognized IP address and MAC address) can generate ARP packets into the network. ARP security is applied to both dynamic and static DHCP snooping entries. For static DHCP entries without a MAC address defined, ARP security compares only the IP address details.

MAC Address Verification

When MAC address verification is enabled, the switch forwards DHCP packets received on untrusted ports only if the source MAC address and client hardware address match. MAC address verification is enabled by default.

DHCP Snooping Violations

Packets violating DHCP snooping or ARP security checks (if these are enabled) are automatically dropped. The switch can also be configured to send SNMP notifications (atDhcpsnTrap and atArpsecTrap), to generate log messages, or to shut down the link on which the packet was received.

If the switch is configured to send notifications for DHCP snooping or ARP security violations, the rate is limited to one notification per second. If there are any further violations within a second, no notifications are sent for them. After one second, the switch only sends further notifications if the source MAC address and/or the violation reason are different from previous notifications. (If log messages are also generated for ARP security and DHCP snooping violations, you can see a record of all violations in the log, even if notifications were not sent for all of them.)

Interactions with Other Features

DHCP snooping interacts with other switch features as follows:

- **Ports in trunk mode**

  The DHCP snooping feature is enabled or disabled per VLAN, and several of the related configuration settings are applied to ports. If there are multiple VLANs on a port, all the VLANs will be subject to the same per-port settings.

- **DHCP relay**

  The switch cannot use DHCP snooping to filter IP traffic from a DHCP relay device.

  DHCP snooping (**service dhcp-snooping** command on page 80.23) and the DHCP relay agent option (**ip dhcp-relay agent-option** command on page 90.14) cannot both be enabled on the switch at the same time.

- **DHCP snooping can be configured with port provisioning**.

- **Authentication**

  DHCP snooping cannot be enabled on a switch that is configured for web authentication (**auth-web enable** command on page 65.35), roaming authentication (**auth roaming enable** command on page 65.19, **auth roaming disconnected** command on page 65.17), or guest VLAN authentication (**auth guest-vlan** command on page 65.9), or vice versa.

- **Stacking**

  If DHCP snooping is enabled in a stack, the DHCP snooping database and its backup file are automatically synchronized across all stack members, so that a new stack master can reinstate this database.

- **Link aggregators**

  DHCP snooping can operate over switch ports, and over static and dynamic (LACP) link aggregators (channel groups). If a switch port is added to an aggregator, DHCP snooping configuration is applied to the aggregator; configuration of the original switch port is not preserved. If the switch port is then removed from the aggregator, it returns to default DHCP snooping settings.

- **Private VLANs**

  Private VLANs are not supported for DHCP snooping.

Configuration

This section provides a general configuration procedure for DHCP snooping.

Configure DHCP Snooping

Note that if a port in trunk mode has multiple VLANs attached, then the DHCP snooping configuration settings for the port apply to all the VLANs.

Table 79-2: General configuration procedure for DHCP snooping

Step 1. Enable DHCP snooping		
1.	`awplus#` `configure terminal`	Enter Global Configuration mode.
2.	`awplus(config)#` `service dhcp-snooping`	Enable DHCP snooping on the switch. Default: disabled
3.	`awplus(config)#` `interface <vid-list>`	Enter Interface Configuration mode for the VLANs to enable DHCP snooping on.
4.	`awplus(config-if)#` `ip dhcp snooping`	Enable DHCP snooping on these VLANs. Default: disabled
5.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
6.	`awplus(config-if)#` `interface <port-list>`	Enter Interface Configuration mode for ports connected to the trusted network. The port(s) connected to the DHCP server(s) must be configured as trusted ports.
7.	`awplus(config-if)#` `ip dhcp snooping trust`	Set these ports to be trusted ports. Default: untrusted
8.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
9.	`awplus(config)#` `interface <port-list>`	If you want to allow more than one DHCP lease for any ports, enter Interface Configuration mode for the required ports. The default is likely to be suitable for edge ports; on an aggregation switch, you may need to increase the maximum number of leases for ports connected to other switches and/or for multiple VLANs. Note that you cannot change this setting once DHCP snooping ACLs are attached to these interfaces.

Table 79-2: General configuration procedure for DHCP snooping(cont.)

10.	`awplus(config-if)#` `ip dhcp snooping max-bindings <0-520>`	Change the maximum number of leases for these ports. Default: 1
11.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
Step 2. Configure DHCP filtering		
12.	`awplus(config)#` `access-list hardware <name>`	Create a hardware access list, and enter Hardware Access List Configuration mode to configure it. See the **access-list hardware (named)** command on page 58.18.

13.	`awplus(config-ip-hw-acl)#`
	`[<seqnum>] permit ip dhcpsnooping any`
	`[<seqnum>] deny ip any any`
		Configure the hardware access list to permit traffic with *source IP address* matching valid entries in the DHCP snooping database, and to deny other traffic. (The last filter applied to the ports by any access list must be the filter that denies all other traffic.)
		OR

	`awplus(config-ip-hw-acl)#`
	`[<seqnum>] permit mac dhcpsnooping any`
	`[<seqnum>] deny mac any any`
		Configure the hardware access list to permit traffic with *source MAC address* matching valid entries in the DHCP snooping database, and to deny other traffic. (The last filter applied to the ports by any access list must be the filter that denies all other traffic.)
		See the **(access-list hardware IP protocol filter)** command on page 58.23.

14.	`awplus(config-ip-hw-acl)#` `exit`	Return to Global Configuration mode.
15.	`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for the ports to add the DHCP snooping access list to. Typically this would be all untrusted ports.
16.	`awplus(config-if)#` `access-group <name>`	Add the hardware-based access list(s) to these ports. The *name* in this command is the name of the access list specified in **Step 12**.
17.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.

Table 79-2: General configuration procedure for DHCP snooping(cont.)

Step 3. Configure ARP security		
18.	`awplus(config)#` `interface <vid-list>`	Enter Interface Configuration mode for the VLANs to enable ARP security on. Default: disabled
19.	`awplus(config-if)#` `arp security`	Enable ARP security on particular VLANs if required. On untrusted ports, ARP security forwards ARP packets that have a source IP address and MAC address matching a dynamic entry in the DHCP snooping database, or an IP address matching a static entry. It drops other ARP packets, and treats them as ARP security violations. Default: disabled
20.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
Step 4. Configure DHCP Relay Agent Option 82		
21.	`awplus(config)#` `no ip dhcp snooping agent-option`	If you do not want the switch to insert DHCP Relay Agent Option 82 information into DHCP packets received on untrusted ports, or to remove this information from DHCP packets transmitted on untrusted ports, disable the DHCP Relay Agent Option 82 agent option. Default: enabled if DHCP snooping is enabled.
22.	`awplus(config)#` `ip dhcp snooping agent-option allow-untrusted`	If there are edge switches that add the DHCP Relay Agent Option 82 information to DHCP packets, and that are connected to untrusted ports on this switch, you may wish to enable this switch to forward these packets, and the associated DHCP reply packets, without changing the DHCP Relay Agent Option 82 information in them. Default: disabled.
23.	`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for one or more ports to add a Subscriber ID for.
24.	`awplus(config-if)#` `ip dhcp snooping subscriber-id [<sub-id>]`	Add the Subscriber ID for these ports. The Subscriber ID is included in DHCP Relay Agent Option 82 information. Default: no Subscriber ID.

Table 79-2: General configuration procedure for DHCP snooping(cont.)

25.	`awplus(config)#` `interface <interface-list>`	Enter Interface Configuration mode for one or more VLANs to add a Circuit ID for.
26.	`awplus(config-if)#` `ip dhcp snooping agent-option circuit-id vlantriplet`	Specify the Circuit ID for the VLAN or group of VLANs as the VLAN ID and port number. Default: VLAN ID and Ifindex number.
27.	`awplus(config)#` `interface <interface-list>`	Enter Interface Configuration mode for one or more VLANs to add a Remote ID for.
28.	`awplus(config-if)#` `ip dhcp snooping agent-option remote-id <remote-id>`	Specify the Remote ID for the VLAN or group of VLANs as an alphanumeric (ASCII) string, 1 to 63 characters in length. Default: the switch's MAC address.
29.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
Step 5. Configure MAC address verification		
30.	`awplus(config)#` `no ip dhcp snooping verify mac-address`	If not required, disable MAC address verification. Default: enabled
Step 6. Configure the DHCP snooping database		
31.	`awplus(config)#` `ip dhcp snooping database {nvs|flash|usb}`	If required, change the location of the file to which the switch writes the dynamic entries from the DHCP snooping database. Default: nvs (non-volatile storage)
32.	`awplus(config)#` `no ip dhcp snooping delete-by-client`	By default, the switch deletes DHCP lease entries from the DHCP snooping database when it receives matching DHCP release messages. Disable these deletions if required, so that lease entries remain in the database until they expire. Default: enabled—entries are deleted when leases are released.
33.	`awplus(config)#` `ip dhcp snooping delete-by-linkdown`	If required, set the switch to delete dynamic entries from the DHCP snooping database when their ports go down. Default: disabled—entries remain if links go down.

Table 79-2: General configuration procedure for DHCP snooping(cont.)

34.	`awplus(config)#` `ip source binding <ipaddr> [<macaddr>] vlan <vid> interface <port>`	You can actively add, modify, or remove static entries from the DHCP snooping database.
35.	`awplus#` `ip dhcp snooping binding <ipaddr> [<macaddr>] vlan <vid> interface <port> expiry <expiry-time>`	You can actively add or remove dynamic entries from the DHCP snooping database. These changes affect the current database and backup file, but are not stored in the running configuration.
Step 7. Configure violation actions		
36.	`awplus(config)#` `interface <port-list>`	Enter Interface Configuration mode for the ports for which you want to configure actions in response to DHCP snooping or ARP security violations.

37.	`awplus(config-if)#`
	`ip dhcp snooping violation {log|trap|link-down} ...`
	`arp security violation {log|trap|link-down} ...`
		If required, set the switch to generate an SNMP notification (trap), to generate a log message, and/or to block traffic on the port on which a DHCP snooping and/or ARP security violation is detected.
		Default: By default, if a packet does not match the DHCP snooping and ARP security restrictions, the packet is dropped, but no other action is taken.

38.	`awplus(config-if)#` `exit`	Return to Global Configuration mode.
39.	`awplus(config)#` `snmp-server enable trap dhcpsnooping`	In order to send SNMP notifications: ■ set the action for violations to trap (**Step 37**) ■ configure SNMP—see **Chapter 94, SNMP Commands** ■ set the SNMP server to enable DHCP snooping notifications (by default notifications are disabled on the SNMP server). The port connecting the switch to the SNMP manager should be set as a trusted port (**Step 7 on page 79.10**).
40.	`awplus(config)#` `exit`	Return to Privileged Exec mode.

Table 79-2: General configuration procedure for DHCP snooping(cont.)

Step 8.	**Check the configuration**	

41.	`awplus#`
	`show ip dhcp snooping`
	`show ip dhcp snooping interface [<port-list>]`
	`show ip dhcp snooping acl`
	`show arp security`
	`show arp security interface [<port-list>]`
	`show running-config dhcp`
		Check the DHCP snooping configuration.

Step 9.	**Troubleshooting DHCP snooping**	
42.	`awplus#` `show ip dhcp snooping binding`	Check all entries in the DHCP snooping database.
43.	`awplus#` `show ip source binding`	Check the static entries in the DHCP snooping database.

44.	`awplus#`
	`show ip dhcp snooping statistics [detail] [interface <interface-list>]`
	`clear ip dhcp snooping statistics [interface <port-list>]`
		Check DHCP snooping statistics.

45.	`awplus#`
	`show arp security statistics [detail] [interface <port-list>]`
	`clear arp security statistics [interface <port-list>]`
		Check ARP security statistics.

46.	`awplus#`
	`debug ip dhcp snooping {all|acl|db|packet [detail]}`
	`show debugging ip dhcp snooping`
	`debug arp security`
	`show debugging arp security`
		Enable debug output for DHCP snooping and/or ARP security.

47.		If you have not already set the switch to log DHCP snooping and ARP security violations, you can do this for troubleshooting purposes. See **Step 37 on page 79.14**.
48.	`awplus#` `show log`	Display the contents of the buffered log, including any DHCP snooping log and debug messages. (See also **Chapter 12, Logging Commands**.)

Disabling DHCP Snooping

If you disable DHCP snooping on the whole switch (**no service dhcp-snooping** command on page 80.23), all the DHCP snooping configuration is removed, except for the Access Control Lists (ACL). Any ACLs on a port that permit traffic matching DHCP snooping entries and block other traffic, will block all traffic if DHCP snooping is disabled on the port. If you disable DHCP snooping either on the whole switch or on particular VLANs (**no ip dhcp snooping** command on page 80.8), you must also remove any DHCP snooping ACLs from the ports to maintain connectivity (**no access-group** command on page 58.4).

Related Features

In addition to configuring DHCP snooping as described in **Table 79-2**, consider whether you also need to configure the following:

- VLANs—see **Chapter 18, VLAN Introduction** and **Chapter 19, VLAN Commands**
- Additional ACL filters—see **Chapter 57, Access Control Lists Introduction** and **Chapter 59, IPv4 Software Access Control List (ACL) Commands**
- QoS—see **Chapter 62, Quality of Service (QoS) Introduction** and **Chapter 63, QoS Commands**
- SNMP—**Chapter 93, SNMP Introduction** and **Chapter 94, SNMP Commands**

Chapter 80: DHCP Snooping Commands

Command List 80.2
arp security 80.2
arp security violation 80.3
clear arp security statistics 80.4
clear ip dhcp snooping binding 80.5
clear ip dhcp snooping statistics 80.6
debug arp security 80.6
debug ip dhcp snooping 80.7
ip dhcp snooping 80.8
ip dhcp snooping agent-option 80.9
ip dhcp snooping agent-option allow-untrusted 80.10
ip dhcp snooping agent-option circuit-id vlantriplet 80.11
ip dhcp snooping agent-option remote-id 80.12
ip dhcp snooping binding 80.13
ip dhcp snooping database 80.14
ip dhcp snooping delete-by-client 80.15
ip dhcp snooping delete-by-linkdown 80.16
ip dhcp snooping max-bindings 80.17
ip dhcp snooping subscriber-id 80.18
ip dhcp snooping trust 80.19
ip dhcp snooping verify mac-address 80.20
ip dhcp snooping violation 80.21
ip source binding 80.22
service dhcp-snooping 80.23
show arp security 80.25
show arp security interface 80.26
show arp security statistics 80.27
show debugging arp security 80.29
show debugging ip dhcp snooping 80.30
show ip dhcp snooping 80.31
show ip dhcp snooping acl 80.32
show ip dhcp snooping agent-option 80.34
show ip dhcp snooping binding 80.36
show ip dhcp snooping interface 80.37
show ip dhcp snooping statistics 80.39
show ip source binding 80.42

Command List

This chapter gives detailed information about the commands used to configure DHCP snooping. For detailed descriptions of related ACL commands, see **Chapter 58, IPv4 Hardware Access Control List (ACL) Commands**. For more information about DHCP snooping, see **Chapter 79, DHCP Snooping Introduction and Configuration**.

DHCP snooping can operate on static link aggregators (e.g. sa2) and dynamic link aggregators (e.g. po2), as well as on switch ports (e.g. port1.0.2).

arp security

Use this command to enable ARP security on untrusted ports in the VLANs, so that the switch only responds to/forwards ARP packets if they have recognized IP and MAC source addresses.

Use the **no** variant of this command to disable ARP security on the VLANs.

Syntax

```
arp security
no arp security
```

Default Disabled

Mode Interface Configuration (VLANs)

Usage Enable ARP security to provide protection against ARP spoofing. DHCP snooping must also be enabled on the switch (**service dhcp-snooping** command on page 80.23), and on the VLANs (**ip dhcp snooping** command on page 80.8).

Example To enable ARP security on VLANs 2 to 4, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# arp security
```

Related Commands

arp security violation
show arp security
show arp security interface
show arp security statistics

arp security violation

Use this command to specify an additional action to perform if an ARP security violation is detected on the ports. ARP security must also be enabled (**arp security** command on page 80.2).

Use the **no** variant of this command to remove the specified action, or all actions. Traffic violating ARP security will be dropped, but no other action will be taken.

Syntax

```
arp security violation {log|trap|link-down} ...
no arp security violation [log|trap|link-down] ...
```

Parameter	Description
`log`	Generate a log message. To display these messages, use the **show log** command on page 12.40.
`trap`	Generate an SNMP notification (trap). To send SNMP notifications, SNMP must also be configured, and DHCP snooping notifications must be enabled using the **snmp-server enable trap** command on page 94.18. Notifications are limited to one per second and to one per source MAC and violation reason. Additional violations within a second of a notification being sent will not result in further notifications. Default: disabled.
`link-down`	Shut down the port that received the packet. Default: disabled.

Default When the switch detects an ARP security violation, it drops the packet. By default, it does not perform any other violation actions.

Mode Interface Configuration (switch ports, static or dynamic aggregated links)

Usage When the switch detects an ARP security violation on an untrusted port in a VLAN that has ARP security enabled, it drops the packet. This command sets the switch to perform additional actions in response to ARP violations.

If a port has been shut down in response to a violation, to bring it back up again after any issues have been resolved, use the **no shutdown** command on page 14.15.

Example To send SNMP notifications for ARP security violations on ports 1.0.1 to 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap dhcpsnooping
awplus(config)# interface port1.0.1-port1.0.6
awplus(config-if)# arp security violation trap
```

Related Commands

arp security
show arp security interface
show arp security statistics
show log
snmp-server enable trap

clear arp security statistics

Use this command to clear ARP security statistics for the specified ports, or for all ports.

Syntax

```
clear arp security statistics [interface <port-list>]
```

Parameter	Description
<port-list>	The ports to clear statistics for. If no ports are specified, statistics are cleared for all ports. The ports may be switch ports, or static or dynamic link aggregators.

Mode

Privileged Exec

Example

To clear statistics for ARP security on interface port1.0.1, use the command:

```
awplus# clear arp security statistics interface port1.0.1
```

Related Commands

arp security violation
show arp security
show arp security statistics

clear ip dhcp snooping binding

Use this command to remove one or more DHCP Snooping dynamic entries from the DHCP Snooping binding database. If no options are specified, all entries are removed from the database.

Caution

If you remove entries from the database for current clients, they will lose IP connectivity until they request and receive a new DHCP lease. If you clear all entries, all clients connected to untrusted ports will lose connectivity.

Syntax

```
clear ip dhcp snooping binding [<ipaddr>] [interface <port-list>]
   [vlan <vid-list>]
```

Parameter	Description
<ipaddr>	Remove the entry for this client IP address.
<port-list>	Remove all entries for these ports. The port list may contain switch ports, and static or dynamic link aggregators (channel groups).
<vid-list>	Remove all entries associated with these VLANs.

Mode Privileged Exec

Usage This command removes dynamic entries from the database. Note that dynamic entries can also be deleted by using the **no** variant of the **ip dhcp snooping binding** command on page 80.13.

Dynamic entries can individually restored by using the **ip dhcp snooping binding** command.

To remove static entries, use the **no** variant of the **ip source binding** command on page 80.22.

Example To remove a dynamic lease entry from the DHCP snooping database for a client with the IP address 192.168.1.2, use the command:

```
awplus# clear ip dhcp snooping binding 192.168.1.2
```

Related Commands
ip dhcp snooping binding
ip source binding
show ip dhcp snooping binding

clear ip dhcp snooping statistics

Use this command to clear DHCP snooping statistics for the specified ports, or for all ports.

Syntax `clear ip dhcp snooping statistics [interface <port-list>]`

Parameter	Description
<port-list>	The ports to clear statistics for. If no ports are specified, statistics are cleared for all ports. The port list can contain switch ports, or static or dynamic link aggregators.

Mode Privileged Exec

Example To clear statistics for the DHCP snooping on interface port1.0.1, use the command:

```
awplus# clear ip dhcp snooping statistics interface port1.0.1
```

Related Commands
clear arp security statistics
show ip dhcp snooping
show ip dhcp snooping statistics

debug arp security

Use this command to enable ARP security debugging.

Use the **no** variant of this command to disable debugging for ARP security.

Syntax

```
debug arp security
no debug arp security
```

Default Disabled

Mode Privileged Exec

Example To enable ARP security debugging, use the commands:

```
awplus# debug arp security
```

Related Commands
show debugging arp security
show log
terminal monitor

debug ip dhcp snooping

Use this command to enable the specified types of debugging for DHCP snooping.

Use the **no** variant of this command to disable the specified types of debugging.

Syntax

```
debug ip dhcp snooping {all|acl|db|packet [detail]}
no debug ip dhcp snooping {all|acl|db|packet [detail]}
```

Parameter	Description
`all`	All DHCP snooping debug.
`acl`	DHCP snooping access list debug.
`db`	DHCP snooping binding database debug.
`packet`	DHCP snooping packet debug. For the **no** variant of this command, this option also disables detailed packet debug, if it was enabled.
`detail`	Detailed packet debug.

Default Disabled

Mode Privileged Exec

Example To enable access list debugging for DHCP snooping, use the commands:

```
awplus# debug ip dhcp snooping acl
```

Related Commands

debug arp security
show debugging ip dhcp snooping
show log
terminal monitor

ip dhcp snooping

Use this command to enable DHCP snooping on one or more VLANs.

Use the **no** variant of this command to disable DHCP snooping on the VLANs.

Syntax

```
ip dhcp snooping
no ip dhcp snooping
```

Default DHCP snooping is disabled on VLANs by default.

Mode Interface Configuration (VLANs)

Usage For DHCP snooping to operate on a VLAN, it must:

- be enabled on the particular VLAN by using this command
- be enabled globally on the switch by using the **service dhcp-snooping** command on page 80.23
- have at least one port connected to a DHCP server configured as a trusted port by using the **ip dhcp snooping trust** command on page 80.19

Any ACLs on a port that permit traffic matching DHCP snooping entries and block other traffic, will block all traffic if DHCP snooping is disabled on the port. If you disable DHCP snooping on particular VLANs using this command, you must also remove any DHCP snooping ACLs from the ports to maintain connectivity (**no access-group** command on page 58.4).

Examples To enable DHCP snooping on VLANs 2 to 4, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# ip dhcp snooping
```

To disable DHCP snooping on the switch, use the command:

```
awplus# configure terminal
awplus(config)# interface vlan2-vlan4
awplus(config-if)# no ip dhcp snooping
```

Related Commands

ip dhcp snooping trust
service dhcp-snooping
show ip dhcp snooping

ip dhcp snooping agent-option

Use this command to enable DHCP Relay Agent Option 82 information insertion on the switch. When this is enabled, the switch:

- inserts DHCP Relay Agent Option 82 information into DHCP packets that it receives on untrusted ports
- removes DHCP Relay Agent Option 82 information from DHCP packets that it sends to untrusted ports.

Use the **no** variant of this command to disable DHCP Relay Agent Option 82 insertion.

Syntax

```
ip dhcp snooping agent-option
no ip dhcp snooping agent-option
```

Default DHCP Relay Agent Option 82 insertion is enabled by default when DHCP snooping is enabled.

Mode Global Configuration

Usage DHCP snooping must also be enabled on the switch (**service dhcp-snooping** command on page 80.23), and on the VLANs (**ip dhcp snooping** command on page 80.8).

If a subscriber ID is configured for the port (**ip dhcp snooping subscriber-id** command on page 80.18), the switch includes this in the DHCP Relay Agent Option 82 information it inserts into DHCP packets received on the port.

Example To disable DHCP Relay Agent Option 82 on the switch, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp snooping agent-option
```

Related Commands

ip dhcp snooping
ip dhcp snooping agent-option allow-untrusted
ip dhcp snooping subscriber-id
service dhcp-snooping
show ip dhcp snooping

ip dhcp snooping agent-option allow-untrusted

Use this command to enable DHCP Relay Agent Option 82 information reception on untrusted ports. When this is enabled, the switch accepts incoming DHCP packets that contain DHCP Relay Agent Option 82 information on untrusted ports.

Use the **no** variant of this command to disable DHCP Relay Agent Option 82 information reception on untrusted ports.

Syntax

```
ip dhcp snooping agent-option allow-untrusted
no ip dhcp snooping agent-option allow-untrusted
```

Default Disabled

Mode Global Configuration

Usage If the switch is connected via untrusted ports to edge switches that insert DHCP Relay Agent Option 82 information into DHCP packets, you may need to allow these DHCP packets through the untrusted ports, by using this command.

When this is disabled (default), the switch treats incoming DHCP packets on untrusted ports that contain DHCP Relay Agent Option 82 information as DHCP snooping violations: it drops them and applies any violation action specified by the **ip dhcp snooping violation** command on page 80.21. The switch stores statistics for packets dropped; to display these statistics, use the **show ip dhcp snooping statistics** command on page 80.39.

Example To enable DHCP snooping Option 82 information reception on untrusted ports, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp snooping agent-option allow-untrusted
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping violation
show ip dhcp snooping
show ip dhcp snooping statistics

ip dhcp snooping agent-option circuit-id vlantriplet

Use this command to specify the Circuit ID sub-option of the DHCP Relay Agent Option 82 field as the VLAN ID and port number. The Circuit ID specifies the switch port and VLAN ID that the client-originated DHCP packet was received on.

Use the **no** variant of this command to set the Circuit ID to the default, the VLAN ID and Ifindex (interface number).

Syntax

```
ip dhcp snooping agent-option circuit-id vlantriplet
no ip dhcp snooping agent-option circuit-id
```

Default By default, the Circuit ID is the VLAN ID and Ifindex (interface number).

Mode Interface Configuration for a VLAN interface.

Usage The Circuit ID sub-option is included in the DHCP Relay Agent Option 82 field of forwarded client DHCP packets:

- DHCP snooping Option 82 information insertion is enabled (**ip dhcp snooping agent-option** command on page 80.9; enabled by default), and
- DHCP snooping is enabled on the switch (**service dhcp-snooping**) and on the VLAN to which the port belongs (**ip dhcp snooping**)

Examples To set the Circuit ID to `vlantriplet` for client DHCP packets received on `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ip dhcp snooping agent-option circuit-id
vlantriplet
```

To return the Circuit ID format to the default for `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# no ip dhcp snooping agent-option circuit-id
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping agent-option remote-id
show ip dhcp snooping
show ip dhcp snooping agent-option

ip dhcp snooping agent-option remote-id

Use this command to specify the Remote ID sub-option of the DHCP Relay Agent Option 82 field. The Remote ID identifies the device that inserted the Option 82 information. If a Remote ID is not specified, the Remote ID sub-option is set to the switch's MAC address.

Use the **no** variant of this command to set the Remote ID to the default, the switch's MAC address.

Syntax

```
ip dhcp snooping agent-option remote-id <remote-id>
no ip dhcp snooping agent-option remote-id
```

Parameter	Description
<remote-id>	An alphanumeric (ASCII) string, 1 to 63 characters in length. If the Remote ID contains spaces, it must be enclosed in double quotes. Wildcards are not allowed.

Default The Remote ID is set to the switch's MAC address by default.

Mode Interface Configuration for a VLAN interface.

Usage The Remote ID sub-option is included in the DHCP Relay Agent Option 82 field of forwarded client DHCP packets:

- DHCP snooping Option 82 information insertion is enabled (**ip dhcp snooping agent-option** command on page 80.9; enabled by default), and
- DHCP snooping is enabled on the switch (**service dhcp-snooping**) and on the VLAN to which the port belongs (**ip dhcp snooping**)

Examples To set the Remote ID to `myid` for client DHCP packets received on `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ip dhcp snooping agent-option remote-id
myid
```

To return the Remote ID format to the default for `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# no ip dhcp snooping agent-option remote-id
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping agent-option circuit-id vlantriplet
show ip dhcp snooping
show ip dhcp snooping agent-option

ip dhcp snooping binding

Use this command to manually add a dynamic-like entry (with an expiry time) to the DHCP snooping database. Once added to the database, this entry is treated as a dynamic entry, and is stored in the DHCP snooping database backup file. This command is not stored in the switch's running configuration.

Use the **no** variant of this command to delete a dynamic entry for an IP address from the DHCP snooping database, or to delete all dynamic entries from the database.

Caution **If you remove entries from the database for current clients, they will lose IP connectivity until they request and receive a new DHCP lease. If you clear all entries, all clients connected to untrusted ports will lose connectivity.**

Syntax

```
ip dhcp snooping binding <ipaddr> [<macaddr>] vlan <vid> interface
   <port> expiry <expiry-time>

no ip dhcp snooping binding [<ipaddr>]
```

Parameter	Description
<ipaddr>	Client's IP address.
<macaddr>	Client's MAC address in HHHH.HHHH.HHHH format.
<vid>	The VLAN ID for the entry, in the range 1 to 4094.
<port>	The port the client is connected to. The port can be a switch port, or a static or dynamic link aggregation (channel group).
<expiry-time>	The expiry time for the entry, in the range 5 to 2147483647 seconds.

Mode Privileged Exec

Usage Note that dynamic entries can also be deleted from the DHCP snooping database by using the **clear ip dhcp snooping binding** command on page 80.5.

To add or remove static entries from the database, use the **ip source binding** command on page 80.22.

Example To restore an entry in the DHCP snooping database for a DHCP client with the IP address 192.168.1.2, MAC address 0001.0002.0003, on port1.0.6 of vlan6, and with an expiry time of 1 hour, use the commands:

```
awplus# ip dhcp snooping binding 192.168.1.2 0001.0002.0003
        vlan 6 interface port1.0.6 expiry 3600
```

Related Commands
clear ip dhcp snooping binding
ip source binding
show ip dhcp snooping binding

ip dhcp snooping database

Use this command to set the location of the file to which the dynamic entries in the DHCP snooping database are written. This file provides a backup for the DHCP snooping database.

Use the **no** variant of this command to set the database location back to the default, **nvs**.

Syntax

```
ip dhcp snooping database {nvs|flash|usb}
no ip dhcp snooping database
```

Parameter	Description
`nvs`	The switch checks the database and writes the file to non-volatile storage (NVS) on the switch at 2 second intervals if it has changed.
`flash`	The switch checks the database and writes the file to Flash memory on the switch at 60 second intervals if it has changed.
`usb`	The switch checks the database and writes the file to a USB storage device installed in the switch at 2 second intervals if it has changed.

Default NVS

Mode Global Configuration

Usage In a stack, the backup file is automatically synchronized across all stack members to the location configured. If the backup file is stored on a USB storage device on the stack master, it is only synchronized across stack members that also have USB storage devices installed.

If the location of the backup file is changed by using this command, a new file is created in the new location, and the old version of the file remains in the old location. This can be removed if necessary (hidden file: **.dhcp.dsn.gz**).

Example To set the location of the DHCP snooping database to non-volatile storage on the switch, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp snooping database nvs
```

Related Commands **show ip dhcp snooping**

ip dhcp snooping delete-by-client

Use this command to set the switch to remove a dynamic entry from the DHCP snooping database when it receives a valid DHCP release message with matching IP address, VLAN ID, and client hardware address on an untrusted port, and to discard release messages that do not match an entry in the database.

Use the **no** variant of this command to set the switch to forward DHCP release messages received on untrusted ports without removing any entries from the database.

Syntax

```
ip dhcp snooping delete-by-client
no ip dhcp snooping delete-by-client
```

Default Enabled: by default, DHCP lease entries are deleted from the DHCP snooping database when matching DHCP release messages are received.

Mode Global Configuration

Usage DHCP clients send a release message when they no longer wish to use the IP address they have been allocated by a DHCP server. Use this command to enable DHCP snooping to use the information in these messages to remove entries from its database immediately. Use the **no** variant of this command to ignore these release messages. Lease entries corresponding to ignored DHCP release messages eventually time out when the lease expires.

Examples To set the switch to delete DHCP snooping lease entries from the DHCP snooping database when a matching release message is received, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp snooping delete-by-client
```

To set the switch to forward and ignore the content of any DHCP release messages it receives, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp snooping delete-by-client
```

Related Commands **show ip dhcp snooping**

ip dhcp snooping delete-by-linkdown

Use this command to set the switch to remove a dynamic entry from the DHCP snooping database when its port goes down. If the port is part of an aggregated link, the entries in the database are only deleted if all the ports in the aggregated link are down.

Use the **no** variant of this command to set the switch not to delete entries when ports go down.

Syntax

```
ip dhcp snooping delete-by-linkdown
no ip dhcp snooping delete-by-linkdown
```

Default Disabled: by default DHCP Snooping bindings are not deleted when an interface goes down.

Mode Global Configuration

Usage If this command is enabled in a stack, and the master goes down and is replaced by a new master, entries in the DHCP snooping database for ports on the master are removed, unless they are part of link aggregators that are still up.

Examples To set the switch to delete DHCP snooping lease entries from the DHCP snooping database when links go down, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp snooping delete-by-linkdown
```

To set the switch *not* to delete DHCP snooping lease entries from the DHCP snooping database when links go down, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp snooping delete-by-linkdown
```

Related Commands **show ip dhcp snooping**

ip dhcp snooping max-bindings

Use this command to set the maximum number of DHCP lease entries that can be stored in the DHCP snooping database for each of the ports. Once this limit has been reached, no further DHCP lease allocations made to devices on the port are stored in the database.

Use the **no** variant of this command to reset the maximum to the default, 1.

Syntax

```
ip dhcp snooping max-bindings <0-520>
no ip dhcp snooping max-bindings
```

Parameter	Description
<0-520>	The maximum number of bindings that will be stored for the port in the DHCP snooping binding database. If 0 is specified, no entries will be stored in the database for the port.

Default The default for maximum bindings is 1.

Mode Interface Configuration (port)

Usage The maximum number of leases cannot be changed for a port while there are DHCP snooping Access Control Lists (ACL) associated with the port. Before using this command, remove any DHCP snooping ACLs associated with the ports. To display ACLs used for DHCP snooping, use the **show ip dhcp snooping acl** command on page 80.32.

In general, the default (1) will work well on an edge port with a single directly connected DHCP client. If the port is on an aggregation switch that is connected to an edge switch with multiple DHCP clients connected through it, then use this command to increase the number of lease entries for the port.

If there are multiple VLANs configured on the port, the limit is shared between all the VLANs on this port. For example, the default only allows one lease to be stored for one VLAN. To allow connectivity for the other VLANs, use this command to increase the number of lease entries for the port.

Example To set the maximum number of bindings to be stored in the DHCP snooping database to 10 per port for ports 1.0.1 to 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.6
awplus(config-if)# ip dhcp snooping max-bindings 10
```

Related Commands

access-group
show ip dhcp snooping acl
show ip dhcp snooping interface

ip dhcp snooping subscriber-id

Use this command to set a Subscriber ID for the ports.

Use the **no** variant of this command to remove Subscriber IDs from the ports.

Syntax

```
ip dhcp snooping subscriber-id [<sub-id>]
no ip dhcp snooping subscriber-id
```

Parameter	Description
<sub-id>	The Subscriber ID; an alphanumeric (ASCII) string 1 to 50 characters in length. If the Subscriber ID contains spaces, it must be enclosed in double quotes. Wildcards are not allowed.

Default No Subscriber ID.

Mode Interface Configuration (port)

Usage The Subscriber ID sub-option is included in the DHCP Relay Agent Option 82 field of client DHCP packets forwarded from a port if:

- a Subscriber ID is specified for the port using this command, and
- DHCP snooping Option 82 information insertion is enabled (**ip dhcp snooping agent-option** command on page 80.9; enabled by default), and
- DHCP snooping is enabled on the switch (**service dhcp-snooping**) and on the VLAN to which the port belongs (**ip dhcp snooping**)

Examples To set the Subscriber ID for port 1.0.3 to **room_534**, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# ip dhcp snooping subscriber-id room_534
```

To remove the Subscriber ID from port 1.0.3, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# no ip dhcp snooping subscriber-id
```

Related Commands

ip dhcp snooping agent-option
show ip dhcp snooping interface

ip dhcp snooping trust

Use this command to set the ports to be DHCP snooping trusted ports.

Use the **no** variant of this command to return the ports to their default as untrusted ports.

Syntax

```
ip dhcp snooping trust
no ip dhcp snooping trust
```

Default All ports are untrusted by default.

Mode Interface Configuration (port)

Usage Typically, ports connecting the switch to trusted elements in the network (towards the core) are set as trusted ports, while ports connecting untrusted network elements are set as untrusted. Configure ports connected to DHCP servers as trusted ports.

Example To set switch ports 1.0.1 and 1.0.2 to be trusted ports, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.2
awplus(config-if)# ip dhcp snooping trust
```

Related Commands show ip dhcp snooping interface

ip dhcp snooping verify mac-address

Use this command to verify that the source MAC address and client hardware address match in DHCP packets received on untrusted ports.

Use the **no** variant of this command to disable MAC address verification.

Syntax

```
ip dhcp snooping verify mac-address
no ip dhcp snooping verify mac-address
```

Default Enabled—source MAC addresses are verified by default.

Mode Global Configuration

Usage When MAC address verification is enabled, the switch treats DHCP packets with source MAC address and client hardware address that do not match as DHCP snooping violations: it drops them and applies any other violation action specified by the **ip dhcp snooping violation** command on page 80.21. To bring the port back up again after any issues have been resolved, use the **no shutdown** command on page 14.15.

Example To disable MAC address verification on the switch, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp snooping verify mac-address
```

Related Commands

ip dhcp snooping violation
show ip dhcp snooping
show ip dhcp snooping statistics

ip dhcp snooping violation

Use this command to specify the action the switch will take when it detects a DHCP snooping violation by a DHCP packet on the ports.

Use the **no** variant of this command to disable the specified violation actions, or all violation actions.

Syntax

```
ip dhcp snooping violation {log|trap|link-down} ...
no ip dhcp snooping violation [{log|trap|link-down} ...]
```

Parameter	Description
`log`	Generate a log message. To display these messages, use the **show log** command on page 12.40. Default: disabled.
`trap`	Generate an SNMP notification (trap). To send SNMP notifications, SNMP must also be configured, and DHCP snooping notifications must be enabled using the **snmp-server enable trap** command on page 94.18. Notifications are limited to one per second and to one per source MAC and violation reason. Default: disabled.
`link-down`	Set the port status to link-down. Default: disabled.

Default By default, DHCP packets that violate DHCP snooping are dropped, but no other violation action is taken.

Mode Interface Configuration (port)

Usage If a port has been shut down in response to a violation, to bring it back up again after any issues have been resolved, use the **no shutdown** command on page 14.15.

IP packets dropped by DHCP snooping filters do not result in other DHCP snooping violation actions.

Example To set the switch to send an SNMP notification and set the link status to link-down if it detects a DHCP snooping violation on switch ports 1.0.1 to 1.0.4, use the commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap dhcpsnooping
awplus(config)# interface port1.0.1-port1.0.4
awplus(config-if)# ip dhcp snooping violation trap link-down
```

Related Commands

show ip dhcp snooping interface
show log
snmp-server enable trap

ip source binding

Use this command to add or replace a static entry in the DHCP snooping database.

Use the **no** variant of this command to delete the specified static entry or all static entries from the database.

Syntax

```
ip source binding <ipaddr> [<macaddr>] vlan <vid> interface <port>
no ip source binding [<ipaddr>]
```

Parameter	Description
<ipaddr>	Client's IP address. If there is already an entry in the DHCP snooping database for this IP address, then this command replaces it with the new entry.
<macaddr>	Client's MAC address in HHHH.HHHH.HHHH format.
<vid>	The VLAN ID associated with the entry.
<port>	The port the client is connected to.

Mode Global Configuration

Usage This command removes static entries from the database.

To remove dynamic entries, use the **clear ip dhcp snooping binding** command on page 80.5 or the **no** variant of the **ip dhcp snooping binding** command on page 80.13.

Examples To add a static entry to the DHCP snooping database for a client with the IP address 192.168.1.2, MAC address 0001.0002.0003, on port1.0.6 of vlan6, use the command:

```
awplus# configure terminal
awplus(config)# ip source binding 192.168.1.2 0001.0002.0003
vlan 6 interface port1.0.6
```

To remove the static entry for IP address 192.168.1.2 from the database, use the commands:

```
awplus# configure terminal
awplus(config)# no ip source binding 192.168.1.2
```

To remove all static entries from the database, use the commands:

```
awplus# configure terminal
awplus(config)# no ip source binding
```

Related Commands

clear ip dhcp snooping binding
ip dhcp snooping binding
show ip dhcp snooping binding
show ip source binding

service dhcp-snooping

Use this command to enable the DHCP snooping service globally on the switch. This must be enabled before other DHCP snooping configuration commands can be entered.

Use the **no** variant of this command to disable the DHCP snooping service on the switch. This removes all DHCP snooping configuration from the running configuration, except for any DHCP snooping maximum bindings settings (**ip dhcp snooping max-bindings** command on page 80.17), and any DHCP snooping-based Access Control Lists (ACLs), which are retained when the service is disabled.

Syntax

```
service dhcp-snooping

no service dhcp-snooping
```

Default DHCP snooping is disabled on the switch by default.

Mode Global Configuration

Usage For DHCP snooping to operate on a VLAN, it must be enabled on the switch by using this command, and also enabled on the particular VLAN by using the **ip dhcp snooping** command on page 80.8.

For DHCP snooping to operate on a VLAN, it must:

- be enabled globally on the switch by using this command
- be enabled on the particular VLAN by using the **ip dhcp snooping** command on page 80.8
- have at least one port connected to a DHCP server configured as a trusted port by using the **ip dhcp snooping trust** command on page 80.19

If you disable the DHCP snooping service by using the **no** variant of this command, all DHCP snooping configuration (including ARP security, but excluding maximum bindings and ACLs) is removed from the running configuration, and the DHCP snooping database is deleted from active memory. If you re-enable the service, the switch:

- repopulates the DHCP snooping database from the dynamic lease entries in the database backup file (in NVS by default—see the **ip dhcp snooping database** command on page 80.14). The lease expiry times are updated.

The DHCP snooping service cannot be enabled on a switch that is configured with any of the following features, or vice versa:

- web authentication (**auth-web enable** command on page 65.35)
- roaming authentication (**auth roaming enable** command on page 65.19, **auth roaming disconnected** command on page 65.17)
- guest VLAN authentication (**auth guest-vlan** command on page 65.9).
- DHCP relay agent option (**ip dhcp-relay agent-option** command on page 90.14)

Any ACLs on a port that permit traffic matching DHCP snooping entries and block other traffic, will block all traffic if DHCP snooping is disabled on the port. If you disable DHCP snooping on the switch using this command, you must also remove any DHCP snooping ACLs from the ports to maintain connectivity (**no access-group** command on page 58.4).

Examples To enable DHCP snooping on the switch, use the command:

```
awplus# configure terminal
awplus(config)# service dhcp-snooping
```

To disable DHCP snooping on the switch, use the command:

```
awplus# configure terminal
awplus(config)# no service dhcp-snooping
```

Related Commands
ip dhcp snooping
ip dhcp snooping database
ip dhcp snooping max-bindings
show ip dhcp snooping

show arp security

Use this command to display ARP security configuration.

Syntax `show arp security`

Mode User Exec and Privileged Exec

Example To display ARP security configuration on the switch use the command:

```
awplus# show arp security
```

Figure 80-1: Example output from the show arp security command

```
awplus# show arp security

ARP Security Information:
  Total VLANs enabled ..... 2
  Total VLANs disabled ..... 11
  vlan1 ..... Disabled
  vlan2 ..... Disabled
  vlan3 ..... Disabled
  vlan4 ..... Disabled
  vlan5 ..... Disabled
  vlan100 ..... Disabled
  vlan101 ..... Disabled
  vlan102 ..... Disabled
  vlan103 ..... Disabled
  vlan104 ..... Disabled
  vlan105 ..... Enabled
  vlan1000 ..... Disabled
  vlan1001 ..... Enabled
```

Table 80-1: Parameters in the output from the show arp security command

Parameter	Description
`Total VLANs enabled`	The number of VLANs that have ARP security enabled.
`Total VLANs disabled`	The number of VLANs that have ARP security disabled.

Related Commands
arp security
show arp security interface
show arp security statistics

show arp security interface

Use this command to display ARP security configuration for the specified ports or all ports.

Syntax `show arp security interface [<port-list>]`

Parameter	Description
<port-list>	The ports to display ARP security information about. The port list can include switch ports, and static or dynamic aggregated links.

Mode User Exec and Privileged Exec

Example To display ARP security configuration for ports, use the command:

```
awplus# show arp security interface
```

Figure 80-2: Example output from the show arp security interface command

```
awplus#show arp security interface

Arp Security Port Status and Configuration:

  Port: Provisioned ports marked with brackets, e.g. (portx.y.z)
  KEY:  LG = Log
        TR = Trap
        LD = Link down

Port         Action
------------------------------
port1.0.1    -- -- --
port1.0.2    -- -- --
port1.0.3    LG TR LD
port1.0.4    LG -- --
port1.0.5    LG -- --
port1.0.6    LG TR --
port1.0.7    LG -- LD
...
```

Table 80-2: Parameters in the output from the show arp security interface command

Parameter	Description
`Action`	The action the switch takes when it detects an ARP security violation on the port.
`Port`	The port. Parentheses indicate that ports are configured for provisioning.
`LG, Log`	Generate a log message
`TR, Trap`	Generate an SNMP notification (trap).
`LD, Link down`	Shut down the link.

Related Commands
arp security violation
show arp security
show arp security statistics
show log
snmp-server enable trap

show arp security statistics

Use this command to display ARP security statistics for the specified ports or all ports.

Syntax `show arp security statistics [detail] [interface <port-list>]`

Parameter	Description
`detail`	Display detailed statistics.
`interface <port-list>`	Display statistics for the specified ports.

Mode User Exec and Privileged Exec

Example To display the brief statistics for the ARP security, use the command:

```
awplus# show arp security statistics
```

Figure 80-3: Example output from the show arp security statistics command

```
awplus# show arp security statistics

DHCP Snooping ARP Security Statistics:

              In          In
 Interface    Packets     Discards
---------------------------------
 port1.0.3    20          20
 port1.0.4    30          30
 port1.0.12   120         0
```

Table 80-3: Parameters in the output from the show arp security statistics command

Parameter	Description
`Interface`	A port name. Parentheses indicate that ports are configured for provisioning.
`In Packets`	The total number of incoming ARP packets that are processed by DHCP Snooping ARP Security
`In Discards`	The total number of ARP packets that are dropped by DHCP Snooping ARP Security.

Figure 80-4: Example output from the show arp security statistics detail command

```
awplus#show arp security statistics detail

DHCP Snooping ARP Security Statistics:

Interface ..................... port1.0.3
  In Packets .................. 20
  In Discards ................. 20
    No Lease .................. 20
    Bad Vlan .................. 0
    Bad Port .................. 0
    Source Ip Not Allocated .... 0

Interface ..................... port1.0.4
  In Packets .................. 30
  In Discards ................. 30
    No Lease .................. 30
    Bad Vlan .................. 0
    Bad Port .................. 0
    Source Ip Not Allocated .... 0

Interface ..................... port1.0.12
  In Packets .................. 120
  In Discards ................. 0
    No Lease .................. 0
    Bad Vlan .................. 0
    Bad Port .................. 0
    Source Ip Not Allocated .... 0
```

Related Commands

- arp security
- arp security violation
- clear arp security statistics
- show arp security
- show arp security interface
- show log

show debugging arp security

Use this command to display the ARP security debugging configuration.

Syntax `show debugging arp security`

Mode User and Privileged Exec

Example To display the debugging settings for ARP security on the switch, use the command:

```
awplus# show debugging arp security
```

Figure 80-5: Example output from the show debugging arp security command

```
awplus# show debugging arp security

ARP Security debugging status:
  ARP Security debugging is off
```

Related Commands **arp security violation**
debug arp security

show debugging ip dhcp snooping

Use this command to display the DHCP snooping debugging configuration.

Syntax `show debugging ip dhcp snooping`

Mode User Exec and Privileged Exec

Example To display the DHCP snooping debugging configuration, use the command:

```
awplus# show debugging ip dhcp snooping
```

Figure 80-6: Example output from the show debugging ip dhcp snooping command

```
awplus# show debugging ip dhcp snooping

DHCP snooping debugging status:
  DHCP snooping debugging is off
  DHCP snooping all debugging is off
  DHCP snooping acl debugging is off
  DHCP snooping binding DB debugging is off
  DHCP snooping packet debugging is off
  DHCP snooping detailed packet debugging is off
```

Related Commands **debug ip dhcp snooping**
show log

show ip dhcp snooping

Use this command to display DHCP snooping global configuration on the switch.

Syntax `show ip dhcp snooping`

Mode User Exec and Privileged Exec

Example To display global DHCP snooping configuration on the switch, use the command:

```
awplus# show ip dhcp snooping
```

Figure 80-7: Example output from the show ip dhcp snooping command

```
DHCP Snooping Information:
  DHCP Snooping service ............. Enabled
  Option 82 insertion ............... Enabled
  Option 82 on untrusted ports ...... Not allowed
  Binding delete by client .......... Disabled
  Binding delete by link down ....... Disabled
  Verify MAC address ................ Disabled
  SNMP DHCP Snooping trap ........... Disabled

DHCP Snooping database:
  Database location ................. nvs
  Number of entries in database ..... 2

DHCP Snooping VLANs:
  Total VLANs enabled ............... 1
  Total VLANs disabled .............. 9
  vlan1 ............. Enabled
  vlan2 ............. Disabled
  vlan3 ............. Disabled
  vlan4 ............. Disabled
  vlan5 ............. Disabled
  vlan100 ........... Disabled
  vlan101 ........... Disabled
  vlan105 ........... Disabled
  vlan1000 .......... Disabled
  vlan1001 .......... Disabled
```

Related Commands
- service dhcp-snooping
- show arp security
- show ip dhcp snooping acl
- show ip dhcp snooping agent-option
- show ip dhcp snooping binding
- show ip dhcp snooping interface

show ip dhcp snooping acl

Use this command to display information about the Access Control Lists (ACL) that are using the DHCP snooping database.

Syntax

```
show ip dhcp snooping acl

show ip dhcp snooping acl [detail|hardware] [interface
   [<interface-list>]]
```

Parameter	Description
`detail`	Detailed DHCP Snooping ACL information.
`hardware`	DHCP Snooping hardware ACL information.
`interface`	ACL Interface information.
<interface-list>	The interfaces to display information about.

Mode User Exec and Privileged Exec

Example To display DHCP snooping ACL information, use the command:

```
awplus# show ip dhcp snooping acl
```

Figure 80-8: Example output from the show ip dhcp snooping acl command

```
awplus#show ip dhcp snooping acl

DHCP Snooping Based Filters Summary:

                       Maximum   Template  Attached
Interface    Bindings  Bindings  Filters   Hardware Filters
-------------------------------------------------------------
 port1.0.1   1         520       0         0
 port1.0.2   1         3         2         6
 port1.0.3   1         2         4         8
 port1.0.4   1         2         7         14
 port1.0.5   0         2         6         12
 port1.0.6   0         1         0         0
 port1.0.7   0         1         0         0
 port1.0.8   0         1         0         0
 port1.0.9   0         1         0         0
 port1.0.10  0         1         0         0
 port1.0.11  0         1         0         0
 port1.0.12  0         1         0         0
(port2.0.1 ) 0         520       0         0
(port2.0.2 ) 0         1         0         0
```

To display DHCP snooping hardware ACL information, use the command:

```
awplus# show ip dhcp snooping acl hardware
```

Figure 80-9: Example output from the show ip dhcp snooping acl hardware command

```
awplus#show ip dhcp snooping acl hardware

DHCP Snooping Based Filters in Hardware:

Interface    Access-list(/ClassMap)        Source IP         Source MAC
-------------------------------------------------------------------------
 port1.0.2   dhcpsn1                       10.10.10.10       aaaa.bbbb.cccc
 port1.0.2   dhcpsn1                       20.20.20.20       0000.aaaa.bbbb
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.2   dhcpsn1                       0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 30.30.30.30       aaaa.bbbb.dddd
 port1.0.3   dhcpsn2/cmap1                 40.40.40.40       0000.aaaa.cccc
 port1.0.3   dhcpsn2/cmap1                 50.50.50.50       0000.aaaa.dddd
 port1.0.3   dhcpsn2/cmap1                 60.60.60.60       0000.aaaa.eeee
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.3   dhcpsn2/cmap1                 0.0.0.0           0000.0000.0000
 port1.0.4   dhcpsn3/cmap2                 70.70.70.70
 port1.0.4   dhcpsn3/cmap2                 80.80.80.80
 port1.0.4   dhcpsn2/cmap1                 70.70.70.70
 port1.0.4   dhcpsn2/cmap1                 80.80.80.80
 port1.0.4   dhcpsn1                       70.70.70.70
 port1.0.4   dhcpsn1                       80.80.80.80
```

To display detailed DHCP snooping ACL information for port 1.0.4, use the command:

```
awplus# show ip dhcp snooping acl detail interface port1.0.4
```

Figure 80-10: Example output from the show ip dhcp snooping acl detail interface command

```
awplus#show ip dhcp snooping acl detail interface port1.0.4

DHCP Snooping Based Filters Information:

 port1.0.4 : Maximum Bindings ........... 2
 port1.0.4 : Template filters ........... 7
 port1.0.4 : Attached hardware filters .. 14
 port1.0.4 : Current bindings ........... 1, 1 free
 port1.0.4    Client 1 ................. 120.120.120.120
 port1.0.4 : Templates: cheese (via class-map: cmap2)
 port1.0.4 :  10 permit ip dhcpsnooping 100.0.0.0/8
 port1.0.4 : Template: dhcpsn2 (via class-map: cmap1)
 port1.0.4 :  10 permit ip dhcpsnooping any
 port1.0.4 :  20 permit ip dhcpsnooping 10.0.0.0/8
 port1.0.4 :  30 permit ip dhcpsnooping 20.0.0.0/8
 port1.0.4 :  40 permit ip dhcpsnooping 30.0.0.0/8
 port1.0.4 : Template: dhcpsn1 (via access-group)
 port1.0.4 :  10 permit ip dhcpsnooping any mac dhcpsnooping abcd.0000.0000 00
 00.ffff.ffff
 port1.0.4 : 20 permit ip dhcpsnooping any
```

Related Commands **access-list hardware (named)**
show access-list (IPv4 Hardware ACLs)

show ip dhcp snooping agent-option

Use this command to display DHCP snooping Option 82 information for all interfaces, a specific interface or a range of interfaces.

Syntax `show ip dhcp snooping agent-option [interface <interface-list>]`

Parameter	Description
`interface`	Specify the interface.
`<interface-list>`	The name of the interface or interface range.

Mode User Exec and Privileged Exec

Examples To display DHCP snooping Option 82 information for all interfaces, use the command:

```
awplus# show ip dhcp snooping agent-option
```

To display DHCP snooping Option 82 information for `port1.0.1`, use the command:

```
awplus# show ip dhcp snooping agent-option interface
        port1.0.1
```

To display DHCP snooping Option 82 information for `vlan1`, use the command:

```
awplus# show ip dhcp snooping agent-option interface vlan1
```

To display DHCP snooping Option 82 information for `port2.0.1`, `port4.0.2` and ports in the range from `port4.0.10` to `port4.0.15`, use the command:

```
awplus# show ip dhcp snooping agent-option interface
        port2.0.1,port4.0.2,port4.0.10-port4.0.15
```

Output

Figure 80-11: Example output from the show ip dhcp snooping agent-option command

```
awplus#show ip dhcp snooping agent-option

DHCP Snooping Option 82 Configuration:

  Key:      C Id = Circuit Id Format
            R Id = Remote Id
            S Id = Subscriber Id

  Option 82 insertion ............... Enabled
  Option 82 on untrusted ports ...... Not allowed

-----------------------------------------------------------------
 vlan1       C Id = vlanifindex
             R Id = Access-Island-01-M1
 vlan2       C Id = vlantriplet
             R Id = Access-Island-01-M1
 vlan3       C Id = vlantriplet
             R Id = Access-Island-01-M3
 vlan4       C Id = vlantriplet
             R Id = 0000.cd28.074c
 vlan5       C Id = vlantriplet
             R Id = 0000.cd28.074c
 vlan6       C Id = vlantriplet
             R Id = 0000.cd28.074c
 port1.0.1   S Id =
 port1.0.2   S Id =
 port1.0.3   S Id = phone_1
 port1.0.4   S Id =
 port1.0.5   S Id = PC_1
 port1.0.6   S Id = phone_2
```

Related Commands

ip dhcp snooping agent-option
ip dhcp snooping agent-option circuit-id vlantriplet
ip dhcp snooping agent-option remote-id
ip dhcp snooping subscriber-id
show ip dhcp snooping
show ip dhcp snooping interface

show ip dhcp snooping binding

Use this command to display all dynamic and static entries in the DHCP snooping binding database.

Syntax `show ip dhcp snooping binding`

Mode User Exec and Privileged Exec

Example To display entries in the DHCP snooping database, use the command:

```
awplus# show ip dhcp snooping binding
```

Figure 80-12: Example output from the show ip dhcp snooping binding command

```
awplus# show ip dhcp snooping binding
DHCP Snooping Bindings:

Client           MAC              Server                          Expires
IP Address       Address          IP Address       VLAN  Port     (sec)       Type
-------------------------------------------------------------------------------
1.2.3.4          aaaa.bbbb.cccc   --               7     1.0.6    Infinite    Stat
1.2.3.6          any              --               4077  1.0.6    Infinite    Stat
1.3.4.5          any              --               1     sa1      Infinite    Stat
111.111.100.101  0000.0000.0001   111.112.1.1      1     1.0.6    4076        Dyna
111.111.101.108  0000.0000.0108   111.112.1.1      1     1.0.6    4084        Dyna
111.111.101.109  0000.0000.0109   111.112.1.1      1     1.0.6    4085        Dyna
111.211.100.101  --               --               1     1.0.2    2147483325  Dyna
111.211.100.109  00b0.0000.0009   111.112.111.111  1     1.0.2    21          Dyna
111.211.101.101  00b0.0000.0101   111.112.111.111  1     1.0.2    214         Dyna

Total number of bindings in database: 9
```

Table 80-4: Parameters in the output from the show ip dhcp snooping binding command

Parameter	Description
`Client IP Address`	The IP address of the DHCP client.
`MAC Address`	The MAC address of the DHCP client.
`Server IP`	The IP address of the DHCP server.
`VLAN`	The VLAN associated with this entry.
`Port`	The port the client is connected to.
`Expires (sec)`	The time in seconds until the lease expires.
`Type`	The source of the entry: ■ Dyna: dynamically entered by snooping DHCP traffic, configured by the **ip dhcp snooping binding** command, or loaded from the database backup file. ■ Stat: added statically by the **ip source binding** command
`Total number of bindings in database`	The total number of dynamic and static lease entries in the DHCP snooping database.

Related Commands
ip dhcp snooping binding
ip dhcp snooping max-bindings
show ip source binding

show ip dhcp snooping interface

Use this command to display information about DHCP snooping configuration and leases for the specified ports, or all ports.

Syntax `show ip dhcp snooping interface [<port-list>]`

Parameter	Description
<port-list>	The ports to display DHCP snooping configuration information for. If no ports are specified, information for all ports is displayed.

Mode User Exec and Privileged Exec

Example To display DHCP snooping information for all ports, use the command:

```
awplus# show ip dhcp snooping interface
```

Figure 80-13: Example output from the show ip dhcp snooping interface command

```
awplus#show ip dhcp snooping interface

DHCP Snooping Port Status and Configuration:

  Port: Provisioned ports marked with brackets, e.g. (portx.y.z)
  Action: LG = Log
          TR = Trap
          LD = Link down

                        Full    Max
 Port         Status     Leases  Leases  Action     Subscriber-ID
-------------------------------------------------------------------------------
 port1.0.1    Untrusted  1       1       LG -- --
 port1.0.2    Untrusted  0       50      LG TR LD   Building 1 Level 1
 port1.0.3    Untrusted  0       50      LG -- --
 port1.0.4    Untrusted  0       50      LG -- --   Building 1 Level 2
 port1.0.5    Untrusted  0       50      LG -- LD   Building 2 Level 1
 port1.0.6    Untrusted  0       1       LG -- --
 port1.0.7    Untrusted  0       1       LG -- --
 port1.0.8    Untrusted  0       1       LG -- --
 port1.0.9    Untrusted  0       1       -- TR --
 port1.0.10   Untrusted  0       1       -- -- LD
 port1.0.11   Trusted    0       1       -- -- --
 port1.0.12   Trusted    0       1       -- -- --
```

Table 80-5: Parameters in the output from the show ip dhcp snooping interface command

Parameter	Description
`Port`	The port interface name.
`Status`	The port status: untrusted (default) or trusted.
`Full Leases`	The number of entries in the DHCP snooping database for the port.
`Max Leases`	The maximum number of entries that can be stored in the database for the port.
`Action`	The DHCP snooping violation actions for the port.
`Subscriber ID`	The subscriber ID for the port. If the subscriber ID is longer than 34 characters, only the first 34 characters are displayed. To display the whole subscriber ID, use the **show running-config dhcp** command on page 7.41.

Related Commands

show ip dhcp snooping
show ip dhcp snooping statistics
show running-config dhcp

show ip dhcp snooping statistics

Use this command to display DHCP snooping statistics.

Syntax

```
show ip dhcp snooping statistics [detail] [interface
   <interface-list>]
```

Parameter	Description
`detail`	Display detailed statistics.
`interface <interface-list>`	Display statistics for the specified interfaces. The interface list can contain switch ports, static or dynamic link aggregators (channel groups), or VLANs.

Mode User Exec and Privileged Exec

Example To show the current DHCP snooping statistics for all interfaces, use the command:

```
awplus# show ip dhcp snooping statistics
```

Figure 80-14: Example output from the show ip dhcp snooping statistics command

```
awplus# show ip dhcp snooping statistics

DHCP Snooping Statistics:

              In            In BOOTP      In BOOTP      In
 Interface    Packets       Requests      Replies       Discards
-------------------------------------------------------------------
 vlan1        444           386           58            223
 port1.0.1    386           386           0             223
 port1.0.2    0             0             0             0
 port1.0.3    0             0             0             0
 port1.0.4    0             0             0             0
 port1.0.5    0             0             0             0
 port1.0.6    58            0             58            0
```

Figure 80-15: Example output from the show ip dhcp snooping statistics detail command

```
awplus# show ip dhcp snooping statistics detail

DHCP Snooping Statistics:

Interface ..................................... port1.0.1, All counters 0
Interface ..................................... port1.0.2, All counters 0
Interface ..................................... port1.0.3, All counters 0
Interface ..................................... port1.0.4
  In Packets .................................. 50
    In BOOTP Requests ......................... 25
    In BOOTP Replies .......................... 25
  In Discards ................................. 1
    Invalid BOOTP Information ................. 0
    Invalid DHCP ACK .......................... 0
    Invalid DHCP Release or Decline ........... 0
    Invalid IP/UDP Header ..................... 0
    Max Bindings Exceeded ..................... 1
    Option 82 Insert Error .................... 0
    Option 82 Received Invalid ................ 0
    Option 82 Received On Untrusted Port ...... 0
    Option 82 Transmit On Untrusted Port ...... 0
    Reply Received On Untrusted Port .......... 0
    Source MAC/CHADDR Mismatch ................ 0
    Static Entry Already Exists ............... 0
Interface ..................................... port1.0.5, All counters 0
Interface ..................................... port1.0.6, All counters 0
```

Table 80-6: Parameters in the output from the show ip dhcp snooping statistics command

Parameter	Description
`Interface`	The interface name.
`In Packets`	The total number of incoming packets that are processed by DHCP Snooping.
`In BOOTP Requests`	The total number of incoming BOOTP Requests.
`In BOOTP Replies`	The total number of incoming BOOTP Replies.
`In Discards`	The total number of incoming packets that have been discarded.
`Invalid BOOTP Information`	Packet contained invalid BOOTP information, such as an invalid BOOTP.OPCode.
`Invalid DHCP ACK`	A DHCP ACK message was discarded, for reasons such as missing Server Option or Lease Option.
`Invalid DHCP Release or Decline`	A DHCP Release or Decline message was discarded, for reasons such as mismatch between received interface and current binding information.
`Invalid IP/UDP Header`	A problem was detected in the IP or UDP header of the packet.
`Max Bindings Exceeded`	Accepting the packet would cause the maximum number of bindings on a port to be exceeded.
`Option 82 Insert Error`	An error occurred while trying to insert DHCP Relay Agent Option 82 information.
`Option 82 Received Invalid`	The DHCP Relay Agent Option 82 information received did not match the information inserted by DHCP Snooping.

Table 80-6: Parameters in the output from the show ip dhcp snooping statistics command(cont.)

Parameter	Description
`Option 82 Received On Untrusted Port`	A packet containing DHCP Relay Agent Option 82 information was received on an untrusted port.
`Option 82 Transmit On Untrusted Port`	A packet containing DHCP Relay Agent Option 82 information was to be sent on an untrusted port.
`Reply Received On Untrusted Port`	A BOOTP reply was received on an untrusted port.
`Source MAC/CHADDR Mismatch`	The L2 Source MAC address of the packet did not match the client hardware address field (BOOTP.CHADDR).
`Static Entry Already Exists`	An entry could not be added as a static entry already exists.

Related Commands

clear ip dhcp snooping statistics
ip dhcp snooping
ip dhcp snooping violation

show ip source binding

Use this command to display static entries in the DHCP snooping database. These are the entries that have been added by using the **ip source binding** command on page 80.22.

Syntax `show ip source binding`

Mode User Exec and Privileged Exec

Example To display static entries in the DHCP snooping database information, use the command:

```
awplus# show ip source binding
```

Figure 80-16: Example output from the show ip source binding command

```
awplus# show ip source binding

IP Source Bindings:

Client       MAC                              Expires
IP Address   Address          VLAN  Port        (sec)      Type
-------------------------------------------------------------------
 1.1.1.1     0000.1111.2222   1     port1.0.1   Infinite   Static
```

Table 80-7: Parameters in the output from the show ip source binding command

Parameter	Description
`Client IP Address`	The IP address of the DHCP client.
`MAC Address`	The MAC address of the DHCP client.
`VLAN`	The VLAN ID the packet is received on.
`Port`	The Layer 2 port name the packet is received on.
`Expires (sec)`	Always infinite for static bindings, or when the leave time in the DHCP message was 0xffffffff (infinite).
`Type`	DHCP Snooping binding type: Static

Related Commands
ip source binding
show ip dhcp snooping binding

Part 6: Network Availability

- Chapter 81 VRRP Introduction and Configuration
- Chapter 82 VRRP Commands
- Chapter 83 EPSR Introduction and Configuration
- Chapter 84 EPSR Commands

Chapter 81: VRRP Introduction and Configuration

VRRP Introduction 81.2
Virtual Router Redundancy Protocol 81.3
VRRP Configuration for IPv4 ..... 81.4
  VRRP Election and Preempt for IPv4 ..... 81.6
VRRP Configuration for IPv6 ..... 81.8
  VRRP Election and Preempt for IPv6 ..... 81.10
VRRP Debugging 81.12
VRRP Configuration Examples 81.13
  VRRP Preferred Master with Backup Configuration 81.13
  VRRP Circuit Failover Configuration 81.16
  VRRPv2 to VRRPv3 Transition Configuration 81.23
  VRRP IPv6 Configuration Example 81.29

VRRP Introduction

This chapter describes the Virtual Router Redundancy Protocol (VRRP) feature provided by the switch, and how to configure the switch to participate in a virtual router. For detailed VRRP command descriptions, examples and output, see **Chapter 82, VRRP Commands**.

One function of a switch is to act as a gateway to the WAN for hosts on a LAN. On larger LANs, two or more switches may act as the gateway, and hosts use a dynamic routing protocol, such as RIP or OSPF, to determine the gateway switch to use as the next hop in order to reach a specific IP destination. However, there are a number of factors, such as administrative or processing overhead or even support for the protocols, which may make it undesirable to use a dynamic routing protocol. One alternative is to use static routing; however, if the statically configured first hop switch fails, the hosts on the LAN are unable to communicate with those on the WAN.

The Virtual Router Redundancy Protocol is defined in RFC 5798 (Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6). It provides a solution to the problem by combining two or more physical switches into a logical grouping called a **virtual router** (VR). The physical switches then operate together to provide a single logical gateway for hosts on the LAN.

A virtual router is configured as the host's gateway and comprises a number of physical routers. The hosts can only see the virtual router so the number of physical routers that make up the virtual router is transparent. If physical routers in the virtual router fail, then traffic to and from the hosts will still be forwarded, so long as there is at least one functioning physical router, no configuration changes will be required by the hosts.

The VRRP virtual router comprises a master router and a number of backup routers. The master router is the router responsible for forwarding packets between the hosts and the remote network. It is also responsible for informing backup router of its presence. Should the master router fail, then one of the backup routers takes over the master router role.

The virtual router uses a special reserved MAC address, which is called VRRP virtual MAC. This MAC address is returned by the master router of the virtual router in any ARP responses relating to the gateway IP address, regardless of which device is acting as the master router. By using this unified MAC address across routers, host maintain connectivity with the remote network if a router fails with a backup taking over a master.

Note If there are PIM-SM routers using VRRP the Bootstrap Router (BSR) function will not work properly.

Virtual Router Redundancy Protocol

The virtual router has a virtual MAC address that is known by all its participating switches or routers. The virtual MAC address is derived from the virtual router identifier - a user-defined value from 1 to 255. At the network level, all hosts on the LAN are configured with a common IP address that is used as the first hop. This IP address is typically owned by the virtual router's preferred individual switch or router. When available, this device performs the duties of the virtual router, and is referred to as the **master**. The switch that owns the IP address associated with the virtual router is referred to as the **preferred master**. When a virtual router is configured so that none of the participating switches owns the IP address, the virtual router has no preferred master.

When a switch takes the role of master for a virtual router, it is responsible for the following:

- Responding to ARP and Neighbor solicitation packets that contain IP addresses associated with the virtual router. The ARP reply or Neighbor response contains the virtual MAC address of the virtual router so that the hosts on the LAN associate the virtual MAC address with their configured first-hop IP address. Note that with VRRPv3 supporting both IPv4 and IPv6, the IP address in this context can be an IPv4 or an IPv6 address.
- Forwarding packets with a destination link layer MAC address equal to the virtual router MAC address.
- The VRRPv3 accept mode is enabled by default in the AlliedWare Plus VRRPv3 implementation. This enables a VRRP master to accept packets addressed to the virtual router IP address even if this IP address is not owned by the VRRP master.
- Broadcasting advertisement packets at regular intervals (at the specified advertisement interval) to inform backup switches that it is still acting as the master switch.

Each of the other switches participating in the virtual router is considered to be a backup switch. A switch can be a member of several different virtual routers on one LAN, but each virtual router must have a unique identifier (VRID). When a switch has the role of backup for a virtual router, it must be able to perform the following tasks:

- Receive advertisement packets from the master and check that the information contained in them is consistent with their own configuration; ignoring and discarding advertisement packets that do not match.
- Assume the role of master for the virtual router if an advertisement packet is not received for a given period (the master-down time), based on the specified advertisement interval (for example: **awplus(config-router)# advertisement-interval 5** will set the advertisement-interval to 5 seconds). The master-down time is approximately three times the advertisement interval.
- Assume the role of master if it receives an advertisement packet from another switch with a lower priority than its own, and if preempt mode is on.

If a VRRP instance is running on a VLAN interface and the VLAN interface goes down, then the VRRP instance, whether it is a VRRP master or a VRRP backup, moves to an INIT state. During the INIT state the VRRP instance on the VLAN interface cannot receive traffic, and will not be active until the VLAN interface is up.

Note When using VRRPv3 with VCStacking, ensure that the VRRPv3 advertisement-interval is configured to a longer time than the VCStacking failover time. If the VRRPv3 advertisement-interval is shorter than the VCStacking failover time, then a VRRPv3 failover will also occur whenever a VCStacking failover occurs. Use seconds not centiseconds to ensure interoperability with VRRPv2.

VRRP Configuration for IPv4

VRRP for IPv4 is disabled by default. Once you have defined a virtual router session, you must enable VRRP to make the session operational for a given interface. You can then enable or disable the virtual router as shown:

To enable VRRP

`awplus(config)#`	
`router vrrp 1 vlan2`	Create a new VRRP session on the router, specify the virtual router ID (VRID) for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`awplus(config-router)#`	
`enable`	Enable the VRRP session on the switch.
`awplus(config-router)#`	
`exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

To disable VRRP

`awplus(config)#`	
`router vrrp 1 vlan2`	Specify an existing VRRP session, specify the virtual router ID (VRID) for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`awplus(config-router)#`	
`disable`	Disable the VRRP session on the switch.
`awplus(config-router)#`	
`exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

A virtual router must be defined on at least two switches before it operates correctly. Use the following steps to configure virtual routing on a switch. Note that this example assumes that VLAN 2 already exists on the switch. See **"Configuring VLANs" on page 18.3**.

To configure virtual routing on a switch

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router vrrp 1 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (vlan2) that will participate in virtual routing.

To configure virtual routing on a switch(cont.)

`awplus(config-router)#` `virtual-ip 10.10.10.50 master`	Set the virtual IP address for the VRRP session. Define the default state (master or backup) of the VRRP router within the virtual router. This sets the default master priority value of 255 without needing to issue a priority command separately.
`awplus(config-router)#` `enable`	Enable the VRRP session on the switch.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

To destroy a virtual router on the LAN, it must be removed from all participating switches. Use the following commands to remove a virtual router so that the switch no longer participates in virtual routing.

To remove the virtual router VRRP 1 from a switch

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `no router vrrp 1 vlan2`	Remove the VRRP session on the switch for the specified interface `vlan2`.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

Alternatively, you can simply disable the virtual router and retain the configuration.

To disable the router and retain the configuration

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router vrrp 1 vlan2`	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.
`awplus(config-router)#` `disable`	Disable the VRRP session on the switch.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration prompt.

VRRP Election and Preempt for IPv4

If the switch that is the current VRRP master becomes unavailable, the master role is taken by the switch with the next highest priority. The priority is a value from 1 to 255, with a default of 100. The value 255 is reserved for the switch that owns the virtual router's IP address. The new master takes over all the responsibilities of the original master.

By default, when a switch becomes available that has a higher priority than the master, this switch takes over as master. This is referred to as **preempt mode** and can be set **on** or **off**. Even with preempt mode **off**, the switch that owns the IP address always becomes the master when available. Preempt mode should be the same for all switches in the virtual router.

If two switches are configured with the same priority and a conflict occurs when they both transition to master simultaneously, the one with the highest IP address has higher priority. Due to timing differences the conflict may not always occur and simply the first switch to respond will become the master.

Hosts on the LAN can continue sending packets to the virtual MAC address they originally associated with the first hop IP address, even though the switch that owns the IP address is not currently available. When the original switch becomes available again, and if it is a preferred switch (i.e. it owns the virtual router IP address) then it resumes the role of master.

Use the following commands to set the priority and preempt mode when you create the virtual router:

To set the priority and preempt mode for VRRP 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router vrrp 1 vlan2`	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.
`awplus(config-router)#` `priority 255`	Set the VRRP priority for the switch
`awplus(config-router)#` `preempt-mode true`	Select the preempt mode for VRRP 1. Note only select preempt mode to true if this has been set to false. Preempt is true by default.
`awplus(config-router)#` `enable`	Enable the VRRP session on the switch.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration prompt

The advertisement interval determines the rate that the master sends its advertisement packets. This rate must be the same value for all switches in the virtual router. The default advertisement interval of 1second can be used for most networks. However, you can modify this interval by using the **advertisement-interval** command, as shown in the following procedure:

To set the advertisement interval to 5 seconds on VRRP1

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config#`	
`router vrrp 1 vlan2`	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.
`awplus(config-router)#`	
`advertisement-interval 5`	Set the advertisement interval to `5` seconds.

VRRP Configuration for IPv6

VRRP for IPv6 is disabled by default. Once you have defined a virtual router session, you must enable VRRP to make the session operational for a given interface. You can then enable or disable the virtual router as shown:

To enable VRRP

`awplus(config)#`	
`router ipv6 vrrp 1 vlan2`	Create a new VRRP session on the router, specify the virtual router ID (VRID) for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`awplus(config-router)#`	
`enable`	Enable the VRRP session on the switch.
`awplus(config-router)#`	
`exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

To disable VRRP

`awplus(config)#`	
`router ipv6 vrrp 1 vlan2`	Specify an existing VRRP session, specify the virtual router ID (VRID) for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`awplus(config-router)#`	
`disable`	Disable the VRRP session on the switch.
`awplus(config-router)#`	
`exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

A virtual router must be defined on at least two switches before it operates correctly. Use the following steps to configure virtual routing on a switch. Note that this example assumes that VLAN 2 already exists on the switch. See **"Configuring VLANs" on page 18.3**.

To configure virtual routing on a switch

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`router ipv6 vrrp 1 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (vlan2) that will participate in virtual routing.

To configure virtual routing on a switch(cont.)

`awplus(config-router)#` `virtual-ipv6 fe80::1 master`	Set the virtual IP address for the VRRP session. Define the default state (master or backup) of the VRRP router within the virtual router. This sets the default master priority value of 255 without needing to issue a priority command separately. Note that `fe80::1` is an IPv6 link-local address. The AlliedWare Plus VRRPv3 implementation supports one IPv6 virtual link local address per virtual router ID. See the **Usage** note for the **virtual-ipv6** command for implementation information about link-local addresses in AlliedWare Plus.
`awplus(config-router)#` `enable`	Enable the VRRP session on the switch.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

To destroy a virtual router on the LAN, it must be removed from all participating switches. Use the following commands to remove a virtual router so that the switch no longer participates in virtual routing.

To remove the virtual router VRRP 1 from a switch

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `no router ipv6 vrrp 1 vlan2`	Remove the VRRP session on the switch for the specified interface `vlan2`.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration mode prompt.

Alternatively, you can simply disable the virtual router and retain the configuration.

To disable the router and retain the configuration

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ipv6 vrrp 1 vlan2`	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.
`awplus(config-router)#` `disable`	Disable the VRRP session on the switch.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration prompt.

VRRP Election and Preempt for IPv6

If the switch that is the current VRRP master becomes unavailable, the master role is taken by the switch with the next highest priority. The priority is a value from 1 to 255, with a default of 100. The value 255 is reserved for the switch that owns the virtual router's IP address. The new master takes over all the responsibilities of the original master.

By default, when a switch becomes available that has a higher priority than the master, this switch takes over as master. This is referred to as **preempt mode** and can be set **on** or **off**. Even with preempt mode **off**, the switch that owns the IP address always becomes the master when available. Preempt mode should be the same for all switches in the virtual router.

If two switches are configured with the same priority and a conflict occurs when they both transition to master simultaneously, the one with the highest IP address has higher priority. Due to timing differences the conflict may not always occur and simply the first switch to respond will become the master.

Hosts on the LAN can continue sending packets to the virtual MAC address they originally associated with the first hop IP address, even though the switch that owns the IP address is not currently available. When the original switch becomes available again, and if it is a preferred switch (i.e. it owns the virtual router IP address) then it resumes the role of master.

Use the following commands to set the priority and preempt mode when you create the virtual router:

To set the priority and preempt mode for VRRP 1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `router ipv6 vrrp 1 vlan2`	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.
`awplus(config-router)#` `priority 255`	Set the VRRP priority for the switch

`awplus(config-router)#` `preempt-mode true`	Select the preempt mode for VRRP 1. Note only select preempt mode to true if this has been set to false. Preempt is true by default.
`awplus(config-router)#` `enable`	Enable the VRRP session on the switch.
`awplus(config-router)#` `exit`	Return to the Global Configuration mode.
`awplus(config)#`	Global Configuration prompt

The advertisement interval determines the rate that the master sends its advertisement packets. This rate must be the same value for all switches in the virtual router. The default advertisement interval of 1second can be used for most networks. However, you can modify this interval by using the **advertisement-interval** command, as shown in the following procedure:

To set the advertisement interval to 5 seconds on VRRP1

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config#` `router ipv6 vrrp 1 vlan2`	Select the VRRP session on the switch, specify the VRID for the session, and specify the interface (`vlan2`) used for virtual routing.
`awplus(config-router)#` `advertisement-interval 5`	Set the advertisement interval to `5` seconds.

VRRP Debugging

VRRP debugging displays data that is useful for troubleshooting. To enable or disable debugging use the following commands:

To select and deselect VRRP debugging

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `debug vrrp [all|events|packet]`	Enable the selected debugging type.
`awplus(config)#` `no debug vrrp [all|events|packet]`	Disable the selected debugging type.

It is important that all switches involved in a virtual router are configured with the same values for the following:

- VRRP virtual router identifier
- IP address
- advertisement interval
- preempt mode
- authentication type
- password

Inconsistent configuration causes advertisement packets to be rejected and the virtual router cannot perform properly.

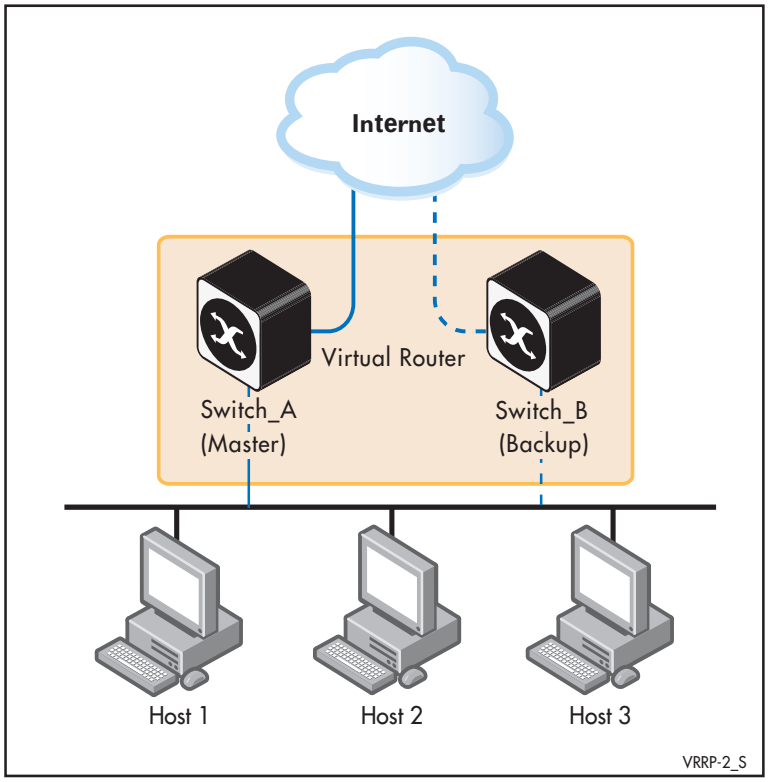
VRRP Configuration Examples

The following examples show how to configure a virtual router in a LAN:

- **VRRP Preferred Master with Backup Configuration**
- **VRRP Circuit Failover Configuration**
- **VRRPv2 to VRRPv3 Transition Configuration**
- **VRRP IPv6 Configuration Example**

VRRP Preferred Master with Backup Configuration

This example show how to configure a basic virtual router with a preferred master and a backup.



Switch_A owns the IP address of the virtual router, and always assumes the role of master whenever it is available. Switch_B is the backup, and assumes the role of master, backing up this IP address if A becomes unavailable.

Step 1: Configure Switch_A

At this point we assume that you have already created VLAN 2 on Switch_A.

See **"Configuring VLANs" on page 18.3** for detailed information about creating VLANs.

Step a. Configure an IP address on VLAN 2

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `hostname Switch_A`	Assign the host name `Switch_A`.
`Switch_A(config)#` `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_A(config-if)#` `ip address 192.168.1.1/24`	Specify the IP address and mask for interface `vlan2`.

Step b. Create the Master Virtual Router

`Switch_A(config)#` `spanning-tree mode stp`	Configure STP for interfaces on `Switch_A`.
`Switch_A(config)#` `router vrrp 1 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`Switch_A(config-router)#` `virtual-ip 192.168.1.1 master`	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
`Switch_A(config-router)#` `enable`	Enable the VRRP session on the router.
`Switch_A(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_A#`	Privileged Exec mode prompt.

Step 2: Configure Switch_B

At this point we assume that you have already created VLAN 2 on Switch_B.

See "Configuring VLANs" on page 18.3 for detailed information about creating VLANs.

Step a. Configure an IP address on VLAN 2

`awplus#`	
`configure terminal`	Enter Global Configuration mode.
`awplus(config)#`	
`hostname Switch_B`	Assign the host name `Switch_B`.
`Switch_B(config)#`	
`interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing.
`Switch_B(config)#`	
`ip address 192.168.1.2/24`	Specify the IP address and mask for interface

Step b. Create the Backup Virtual Router

`Switch_B(config)#`	
`router vrrp 1 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`Switch_B(config-router)#`	
`virtual-ip 192.168.1.1 backup`	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
`Switch_B(config-router)#`	
`enable`	Enable the VRRP session on the router.
`Switch_B(config-router)`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.
`Switch_B(config)#`	
`exit`	Return to the Privileged Exec mode.
`Switch_B#`	Privileged Exec mode prompt.

Commands Used

enable (VRRP)
router vrrp (interface)
virtual-ip

Validation Commands

show vrrp

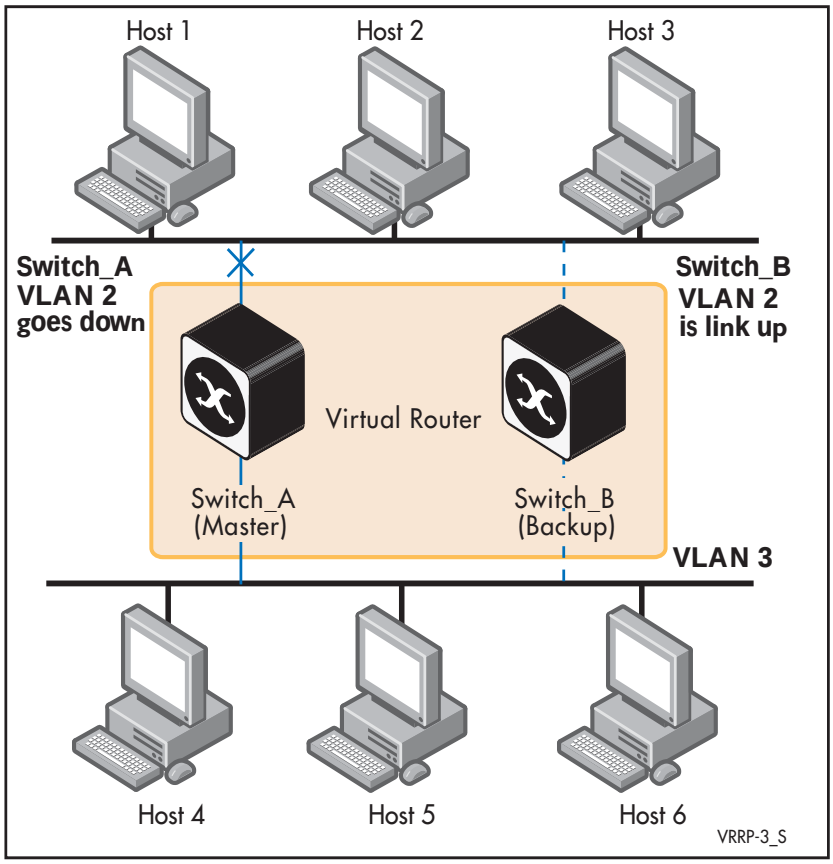
VRRP Circuit Failover Configuration

This example show how to configure a circuit-failover on a virtual router. This example configures redundancy between Switch_A and Switch_B for hosts on VLAN 2 and VLAN 3.

The need for VRRP Circuit Failover arose because VRRPv2 was unable to track the gateway interface status. The AlliedWare Plus VRRP Circuit Failover feature provides a dynamic failover of an entire circuit in the event that one of the members of the group fails.

This introduces the concept of a circuit, where two or more Virtual Routers on a single system are grouped. In the event of a failure occurring a Virtual Router performs the Master to Backup transition and notifies the other Virtual Routers. These are then forced into the Master to Backup transition, so that both incoming and outgoing packets are routed through the same gateway router, eliminating the problem for NAT environments.

To configure VRRP Circuit Failover, each circuit is configured to have a corresponding priority delta value, which is passed to VRRP when a failure occurs. The priority of each Virtual Router on the circuit is decremented by the priority delta value, which causes the Virtual Router Master to Virtual Router Backup transition. In this example, two switches Switch_A and Switch_B are configured as backup routers with different priorities. The priority delta value is configured to be greater that the difference of both the priorities.



Switch_A is configured to have a priority of 100, and Switch_B is configured to have a priority of 90. Switch_A with a greater priority is the Virtual Router Master. The priority delta value is 20, greater than 10 (100 minus 90). On Switch_A, when vlan2 fails, the priority of Switch_A becomes 80 (100 minus 20). Since Switch_B has a greater priority (90) than Switch_A, Switch_B becomes the Virtual Router Master, and routing of packets continues without interruption. When this Virtual Router Backup (Switch_A) is up again, it regains its original priority (100), and becomes the Virtual Router Master again.

See also the **circuit-failover** command description in **Chapter 82, VRRP Commands**.

Step 1: Configure Switch_A

At this point we assume that you have already created VLAN 2 on VR1. See **"Configuring VLANs" on page 18.3** for detailed information about creating VLANs.

Step a. Configure an IP address on VLAN 2

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`hostname Switch_A`	Assign the host name `Switch_A`.
`Switch_A(config)#`	
`interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_A(config-if)#`	
`ip address 192.168.1.1/24`	Specify the IP address and mask for interface `vlan2`.

Step b. Configure an IP address on VLAN 3

`Switch_A(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#`	
`interface vlan3`	Specify the interface (`vlan3`) that will participate in virtual routing to first configure an IP address for `vlan3`.
`Switch_A(config-if)#`	
`ip address 192.168.2.1/24`	Specify the IP address and mask for interface `vlan3`.

Step c. Create the Master Virtual Router

`Switch_A(config)#` `router vrrp 1 vlan2`	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`Switch_A(config-router)#` `virtual-ip 192.168.1.1 backup`	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
`Switch_A(config-router)#` `priority 100`	Set the VRRP priority to 100 as the default priority for a Backup Virtual Router.
`Switch_A(config-router)#` `preempt-mode true`	Set preempt mode to true to specify that the highest priority will own the virtual IP address when there is a failure and will function as the Backup Virtual Router.
`Switch_B(config-router)#` `advertisement-interval 5`	Configure the default value for the advertisement interval. The configurable range for the advertisement interval is 1-10.
`Switch_A(config-router)#` `circuit-failover vlan2 20`	Configure circuit failover to 20 on Switch_A. This configures a priority delta value, greater than the difference of priorities on master and backup routers. This priority delta value is subtracted from the current VR Master Router priority value.
`Switch_A(config-router)#` `enable`	Enable the VRRP session on the router.
`Switch_A(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_A#`	Privileged Exec mode prompt.
`Switch_A#` `copy running-config startup-config`	Copy the contents of the running-configuration to the startup-configuration.

Step 2: Configure Switch_B

At this point we assume that you have already created VLAN2on Switch_B.

See "Configuring VLANs" on page 18.3 for detailed information about creating VLANs.

Step a. Configure an IP address on VLAN 2

awplus# configure terminal	Enter Global Configuration mode.
awplus(config)# hostname Switch_B	Assign the host name Switch_B.
Switch_B(config)# interface vlan2	Specify the interface (vlan2) that will participate in virtual routing.
Switch_B(config-if)# ip address 192.168.1.2/24	Specify the IP address and mask for the interface

Step b. Configure an IP address on VLAN 3

Switch_B(config-if)# exit	Exit the Interface Configuration mode and enter the Global Configuration mode.
Switch_B(config)# interface vlan3	Specify the interface (vlan3) that will participate in virtual routing.
Switch_B(config-if)# ip address 192.168.2.2/24	Specify the IP address and mask for the interface

Step c. Create the Backup Virtual Router

Switch_B(config)# router vrrp 1 vlan2	Create a new VRRP session on the router, specify the VRID for the session, and specify the interface (vlan2) that will participate in virtual routing.
Switch_B(config-router)# virtual-ip 192.168.1.1 backup	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router.
Switch_B(config-router)# priority 90	Set the VRRP priority to 90 (less than 100) because Switch_B is the Backup Virtual Router.
Switch_B(config-router)# preempt-mode true	Set preempt mode to true to specify that the highest priority will own the virtual IP address when there is a failure and will function as the Backup Virtual Router.
Switch_B(config-router)# advertisement-interval 5	Configure the default value for the advertisement interval. The configurable range for the advertisement interval is 1-10.

`Switch_B(config-router)#`	
`enable`	Enable the VRRP session on the router.
`Switch_A(config-router)#`	
`exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#`	
`exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_A#`	Privileged Exec mode prompt.
`Switch_A#`	
`copy running-config startup-config`	Copy the contents of the running-configuration to the startup-configuration.

Commands Used

advertisement-interval
circuit-failover
enable (VRRP)
preempt-mode
priority
router vrrp (interface)
virtual-ip

Validation Commands

show vrrp

VRRPv2 to VRRPv3 Transition Configuration

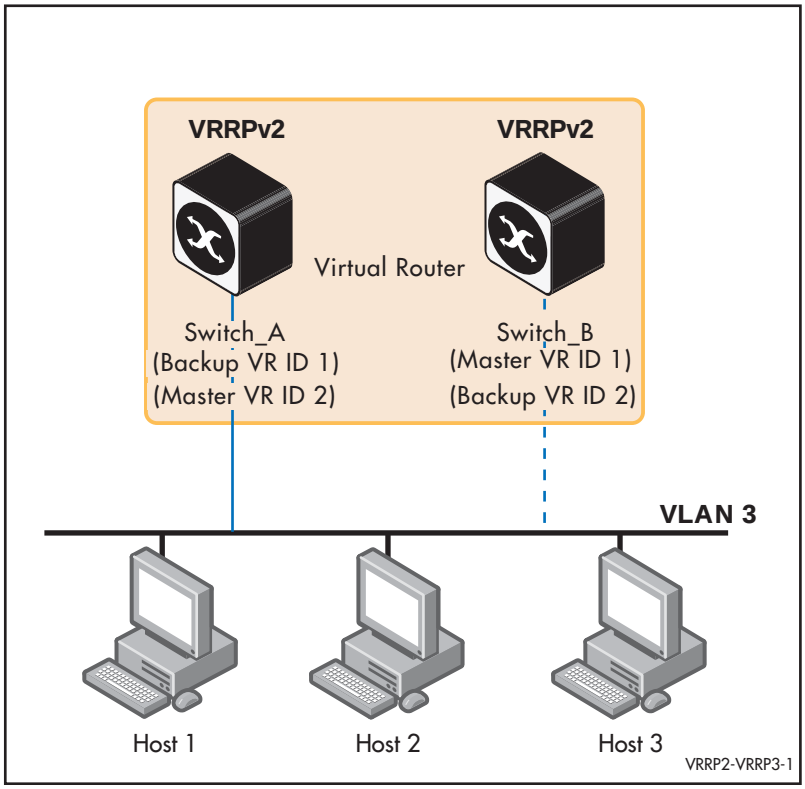
This example shows how to configure the transition from VRRPv2 to VRRPv3 on a virtual router. This example configures VRRPv3 from VRRPv2 on Switch_A and Switch_B for hosts on VLAN 3. See also the **transition-mode** command in **Chapter 82, VRRP Commands**.

Transition mode allows interoperability for IPv4 VRRP instances between VRRPv2 and VRRPv3 virtual routers. RFC 5798 for VRRPv3 allows a VRRPv3 virtual router to send both VRRPv2 and VRRPv3 advertisements. Note that this feature is only for IPv4 interoperability. Note that when **transition-mode** is set to true then VRRPv3 will also accept and process VRRPv2 advertisement packets received should there be a VRRPv2 master on the network.

You should upgrade your VRRPv2 virtual routers to VRRPv3 as a long term solution and only use transition mode for a staged VRRPv2 to VRRPv3 roll out. Transition mode is disabled by default and is enabled by issuing the **transition-mode true** command.

Note that you should ensure that the advertisement interval for a VRRPv2 instance is configured to greater than 1 second (100 centiseconds). If the advertisement interval is configured to less than 1 second (100 centiseconds) then ensure the VRRPv2 virtual router has a lower priority with the **priority** command than the VRRPv2/VRRPv3 virtual router.

When you configure a VRRPv3 instance with **transition-mode true** also configure it as the VRRP master instance, either by configuring it to own the IP address or with a high priority. Also configure the advertisement interval to whole seconds to maintain compatibility with VRRPv2 hosts. All matching VRRP instances should be configured with the same advertisement interval to eliminate master instance contention on VRRP startup as well.



Follow the steps listed over the page to disable VRRPv2 on Switch_A before saving the running and startup configurations then rebooting Switch_A to upgrade to VRRPv3.

Note that after upgrading Switch_A to VRRPv3 you can leave Switch_B running VRRPv2, or you can upgrade Switch_B to VRRPv3. Running VRRPv3 on both is highly recommended. The above illustration shows both Switch_A and Switch_B running VRRPv2 to upgrade.

Follow **Step 1** to upgrade Switch_A from VRRPv2 to VRRPv3 and follow **Step 2** to upgrade Switch_B from VRRPv2 to VRRPv3. You can follow **Step 1** only if Switch_B is not upgraded. Only Switch_A needs transition-mode enabled to upgrade then disabled after upgrading.

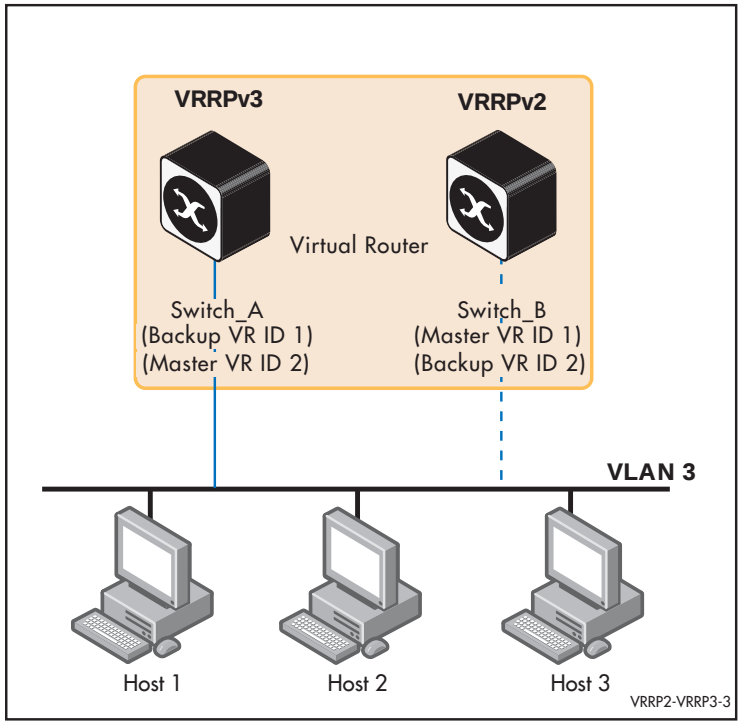
Step 1: Upgrade Switch_A from VRRPv2 to VRRPv3

At this point we assume that you have already copied the current release of AlliedWare Plus with VRRPv3 to flash and only need to make this release the boot version and restart. We also assume that you have already created VLAN 3 on Switch_B. See **"Configuring VLANs" on page 18.3** for detailed information about creating VLAN 3 as needed.

Disable VRRPv2 on Switch_A and enable VRRPv3 on Switch_A

`Switch_A#` `configure terminal`	Enter the Global Configuration mode for `Switch_A`.
`Switch_A(config)#` `router vrrp 1 vlan3`	Select the VRRPv2 backup session on the router, specify the VRID for the session (`1`), and specify the interface (`vlan3`).
`Switch_A(config-router)#` `disable`	Disable the VRRPv2 backup on Switch_A.
`Switch_A(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#` `router vrrp 2 vlan3`	Select the VRRPv2 master session on the router, specify the VRID for the session (`2`), and specify the interface (`vlan3`).
`Switch_A(config-router)#` `disable`	Disable the VRRPv2 master on Switch_A.
`Switch_A(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_A#` `copy running-config startup-config`	Copy the contents of the running-configuration to the startup-configuration.
`Switch_A#` `configure terminal`	Enter the Global Configuration mode.
`Switch_A(config)#` `reload`	Restart `Switch_A` to load the AlliedWare Plus release with VRRPv3 as configured with the **boot system** command earlier.

`Switch_A(config)#` `router vrrp 1 vlan3`	Enter Router Configuration mode for the backup VRRPv3 session on the router, specifying the VRID for the session (`1`), and specifying the interface (`vlan3`).
`Switch_A(config-router)#` `transition-mode true`	Set transition mode to true to turn on transition mode enabling VRRPv2 and VRRPv3 advertisement on VRRPv3.
`Switch_A(config-router)#` `enable`	Enable the VRRPv3 backup on Switch_A.
`Switch_A(config)#` `router vrrp 2 vlan3`	Create a VRRPv3 master session on the router, specify the VRID for the session (`2`), and specify the interface (`vlan3`).
`Switch_A(config-router)#` `transition-mode true`	Set transition mode to true to turn on transition mode enabling VRRPv2 and VRRPv3 advertisement on VRRPv3.
`Switch_A(config-router)#` `enable`	Enable the VRRPv3 master on Switch_A.
`Switch_A(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_A#` `copy running-config startup-config`	Copy the contents of the running-configuration to the startup-configuration.



The above illustration shows Switch_A running VRRPv3 and Switch_B running VRRPv2. Note that Switch_A running VRRPv3 with **transition-mode true** configured sends both VRRPv3 and VRRPv2 advertisements. This is an interim solution for IPv4 VRRPv2 and VRRPv3 interoperability. Only VRRPv3 should be used on both devices for IPv6 use.

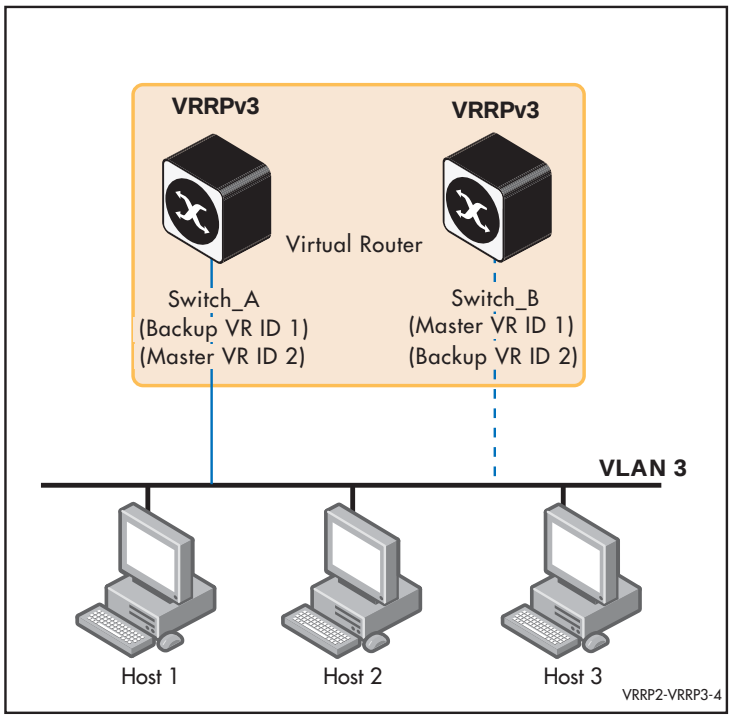
Step 2: Upgrade Switch_B from VRRPv2 to VRRPv3

At this point we assume that you have already copied the current release of AlliedWare Plus with VRRPv3 to flash and only need to make this release the boot version and restart. We also assume that you have already created VLAN 3 on Switch_B. See **"Configuring VLANs" on page 18.3** for detailed information about creating VLAN 3 as needed.

Disable VRRPv2 on Switch_B and enable VRRPv3 on Switch_B

`Switch_B#` `configure terminal`	Enter the Global Configuration mode for `Switch_B`.
`Switch_B(config)#` `router vrrp 1 vlan3`	Select the VRRPv2 master session on the router, specify the VRID for the session (`1`), and specify the interface (`vlan3`).
`Switch_B(config-router)#` `disable`	Disable the VRRPv2 master on Switch_B.
`Switch_B(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_B(config)#` `router vrrp 2 vlan3`	Select the VRRPv2 backup session on the router, specify the VRID for the session (`2`), and specify the interface (`vlan3`).

`Switch_B(config-router)#` `disable`	Disable the VRRPv2 backup on Switch_B.
`Switch_B(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_B(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_B#` `copy running-config startup-config`	Copy the contents of the running-configuration to the startup-configuration.
`Switch_B#` `configure terminal`	Enter the Global Configuration mode.
`Switch_B(config)#` `reload`	Restart `Switch_A` to load the AlliedWare Plus release with VRRPv3 as configured with the **boot system** command earlier.
`Switch_B(config)#` `router vrrp 1 vlan3`	Enter Router Configuration mode for the master VRRPv3 session on the router, specifying the VRID for the session (`1`), and specifying the interface (`vlan3`).
`Switch_B(config-router)#` `enable`	Enable the VRRPv3 master on Switch_B.
`Switch_B(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_B(config)#` `router vrrp 2 vlan3`	Enter Router Configuration mode for the backup VRRPv3 session on the router, specifying the VRID for the session (`2`), and specifying the interface (`vlan3`).
`Switch_B(config-router)#` `enable`	Enable the VRRPv3 backup on Switch_B.
`Switch_B(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_B(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_B#` `copy running-config startup-config`	Copy the contents of the running-configuration to the startup-configuration.



The above illustration shows both Switch_A and Switch_B running VRRPv3. Note that transition mode should be turned off Switch_A once VRRPv3 is active on both to stop Switch_A from continuing to send VRRPv2 and VRRPv3 advertisements to Switch_B.

`Switch_A#`	
`configure terminal`	Enter the Global Configuration mode for `Switch_A`.
`Switch_A(config)#`	
`router vrrp 1 vlan3`	Enter Router Configuration mode for the backup VRRPv3 session on the router, specifying the VRID for the session (`1`), and specifying the interface (`vlan3`).
`Switch_A(config-router)#`	
`transition-mode false`	Set transition mode to false to turn off transition mode now VRRPv3 is on Switch_A and Switch_B.
`Switch_A(config-router)#`	
`exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#`	
`router vrrp 2 vlan3`	Enter Router Configuration mode for the master VRRPv3 session on the router, specifying the VRID for the session (`2`), and specifying the interface (`vlan3`).
`Switch_A(config-router)#`	
`transition-mode false`	Set transition mode to false to turn off transition mode now VRRPv3 is on Switch_A and Switch_B.

Commands Used

disable (VRRP)
enable (VRRP)
router vrrp (interface)
transition-mode

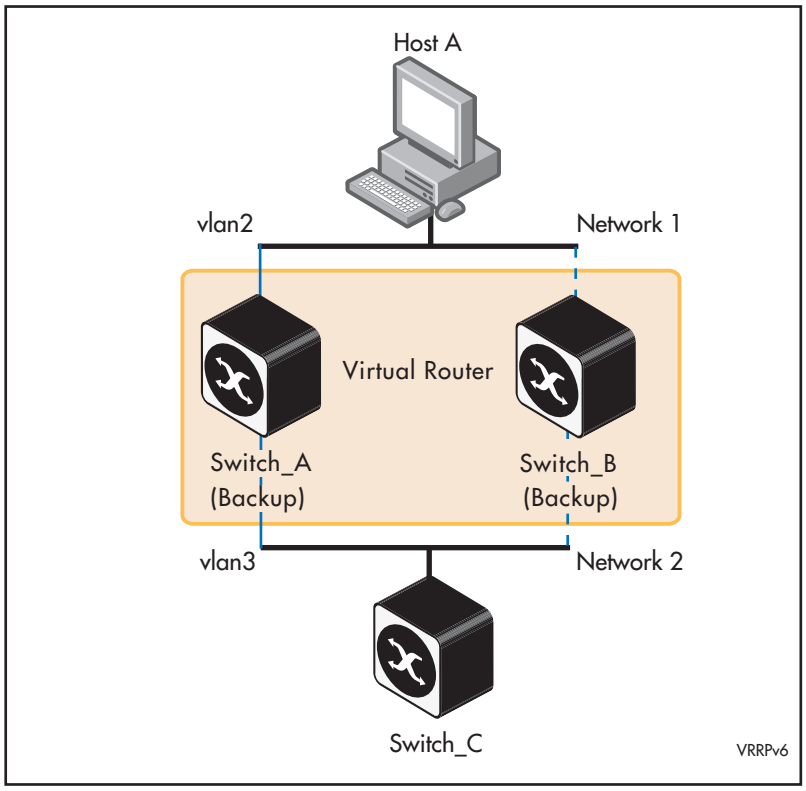
VRRP IPv6 Configuration Example

This section shows a Virtual Router Redundancy Protocol IPv6 (VRRPv3) configuration example. For detailed VRRP command descriptions, configuration command examples and relevant sample show command output, see **Chapter 82, VRRP Commands**.

VRRPv3 eliminates the risk of a single point of failure inherent in a static default routing environment. VRRPv3 specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the devices configured with VRRPv3 on a LAN.

VRRRv3 only allows Master/Non-Owner setup. You can configure the priority of the routers. Router with the higher priority takes the role of master.

In this example, VRRPv3 is enabled on Switch_A, the Backup Virtual Router, and on Switch_B, the Backup Virtual Router. In Switch_A, the static routes are redistributed.



Switch_A owns the IP address of the virtual router, and assumes the role of master because it is configured with a higher priority. Switch_B is the backup, and assumes the role of master, backing up this IP address if A becomes unavailable. No authentication is used for this simple virtual router.

See the sample output following the sample command configuration tables for each device. See relevant VRRP show commands that are useful to validate configurations.

The AlliedWare Plus VRRPv3 implementation supports one IPv6 virtual link local address per virtual router ID. Note that in the command examples, fe80::1 is an IPv6 link-local address. An IPv6 link-local address is used because IPv6 link-local addresses are used by IPv6 ND (Neighbor Discovery). A host's default route to a router points to the IPv6 link-local address, not a specific global IPv6 address for the router. For the host's traffic to switch over a backup router, the IPv6 link-local address of the router is used by VRRPv3.

Step 1: Configure Switch_A (Backup Virtual Router)

At this point we assume that you have already created VLAN 2 and VLAN 3 on Switch_A. See "Configuring VLANs" on page 18.3 for detailed information about creating VLANs.

Step a. Configure IPv6 addresses on VLAN 2 and VLAN 3

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `hostname Switch_A`	Assign a host name to `Switch_A`.
`Switch_A(config)#` `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_A(config-if)#` `ipv6 address 2001:db8:2::2/64`	Specify the IPv6 address and mask for interface `vlan2`.
`Switch_A(config-if)#` `exit`	Return to Global Configuration mode.
`Switch_A(config)#` `interface vlan3`	Specify the interface (`vlan3`) that will participate in virtual routing to first configure an IP address for `vlan3`.
`Switch_A(config-if)#` `ipv6 address 2001:db8:3::2/64`	Specify the IPv6 address and mask for interface `vlan3`.
`Switch_A(config-if)#` `exit`	Return to Global Configuration mode.

Step b. Create the Backup Virtual Router on Switch_A

`Switch_A(config)#` `router ipv6 vrrp 1 vlan2`	Create a new VRRPv3 session on Switch_A, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`Switch_A(config-router)#` `virtual-ipv6 fe80::1 backup`	Set the virtual IP address for the VRRPv3 session. Define the default state of the VRRPv3 router within the virtual router. Note that `fe80::1` is an IPv6 link-local address. The AlliedWare Plus VRRPv3 implementation supports one IPv6 virtual link local address per virtual router ID. See the **Usage** note for the **virtual-ipv6** command for implementation information about link-local addresses in AlliedWare Plus.
`Switch_A(config-router)#` `advertisement-interval 5`	Configure the default value for the advertisement interval. The configurable range for the advertisement interval is 1-10.
`Switch_A(config-router)#` `priority 255`	Configure the default priority value of 255 when the device is the Master Virtual Router.
`Switch_A(config-router)#` `preempt-mode true`	Set the preempt-mode to true for Switch_A. The default preempt mode ensures that the highest priority switch available always takes the master role.
`Switch_A(config-router)#` `enable`	Enable the VRRPv3 session on Switch_A.
`Switch_A(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_A(config)#` `exit`	Return to Privileged Exec mode.
`Switch_A#` `copy running-config startup-config`	Copy the running-config to the startup-config to enable this configuration to execute after restarting Switch_A.

Step 2: Configure Switch_B (Backup Virtual Router)

At this point we assume that you have already created VLAN 2 and VLAN 3 on Switch_B. See **"Configuring VLANs" on page 18.3** for detailed information about creating VLANs.

Step a. Configure IPv6 addresses on VLAN 2 and VLAN 3

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`hostname Switch_B`	Assign a host name to `Switch_B`.
`Switch_B(config)#`	
`interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_B(config-if)#`	
`ipv6 address 2001:db8:2::3/64`	Specify the IPv6 address and mask for interface `vlan2`.
`Switch_B(config-if)#`	
`exit`	Return to Global Configuration mode.
`Switch_B(config)#`	
`interface vlan3`	Specify the interface (`vlan3`) that will participate in virtual routing to first configure an IP address for `vlan3`.
`Switch_B(config-if)#`	
`ipv6 address 2001:db8:3::3/64`	Specify the IPv6 address and mask for interface `vlan3`.
`Switch_B(config-if)#`	
`exit`	Return to Global Configuration mode.

Step b. Create the Backup Virtual Router on Switch_B

`Switch_B(config)#` `router ipv6 vrrp 1 vlan2`	Create a new VRRPv3 session on Switch_B, specify the VRID for the session, and specify the interface (`vlan2`) that will participate in virtual routing.
`Switch_B(config-router)#` `virtual-ipv6 fe80::1 backup`	Set the virtual IP address for the VRRP session. Define the default state of the VRRP router within the virtual router. Note that `fe80::1` is an IPv6 link-local address. The AlliedWare Plus VRRPv3 implementation supports one IPv6 virtual link local address per virtual router ID. See the **Usage** note for the **virtual-ipv6** command for implementation information about link-local addresses in AlliedWare Plus.
`Switch_B(config-router)#` `advertisement-interval 5`	Configure the default value for the advertisement interval. The configurable range for the advertisement interval is 1-10.
`Switch_B(config-router)#` `priority 100`	Configure the priority value of 100 when the device is the Backup Virtual Router.
`Switch_B(config-router)#` `preempt-mode true`	Set the preempt-mode to true for Switch_B. The default preempt mode ensures that the highest priority switch available always takes the master role.
`Switch_B(config-router)#` `enable`	Enable the VRRPv3 session on Switch_B.
`Switch_B(config-router)#` `exit`	Exit the Router Configuration mode and enter the Global Configuration mode.
`Switch_B(config)#` `exit`	Exit the Global Configuration mode and enter the Privileged Exec mode.
`Switch_B#` `copy running-config startup-config`	Copy the running-config to the startup-config to enable this configuration to execute after restarting Switch_B.

Step 3: Configure Host_A

At this point we assume that you have already created VLAN 2 and VLAN 3 on Host_A.

See "Configuring VLANs" on page 18.3 for detailed information about creating VLANs.

Configure an IPv6 address on VLAN 2 and a static IPv6 route to reach Switch_C on VLAN 3

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `hostname Host_A`	Assign the host name to `Host_A`.
`Host_A(config)#` `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Host_A(config-if)#` `ipv6 address 2001:db8:2::1/64`	Specify the IPv6 address and mask for interface `vlan2`.
`Host_A(config-if)#` `exit`	Return to Global Configuration mode.
`Host_A(config)#` `ipv6 route 2001:db8::/64`	Configure an IPv6 static route to reach interface `vlan3` of Host_A.
`Host_A(config)#` `exit`	Return to Privileged Exec mode.
`Host_A#` `copy running-config startup-config`	Copy the running-config to the startup-config to enable this configuration to execute after restarting Host_A.

Step 4: Configure Switch_C

At this point we assume that you have already created VLAN 2 and VLAN 3 on Switch_B. See "Configuring VLANs" on page 18.3 for detailed information about creating VLANs..

Configure IPv6 addresses on VLAN 2 and VLAN 3:

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `hostname Switch_C`	Assign the host name `Switch_C`.
`Switch_C(config)#` `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_C(config-if)#` `ipv6 address 2001:db8:2::4/64`	Specify the IPv6 address and mask for interface `vlan2`.
`Switch_C(config-if)#` `exit`	Return to Global Configuration mode.
`Switch_C(config)#` `interface vlan3`	Specify the interface (`vlan3`) that will participate in virtual routing to first configure an IP address for `vlan3`.
`Switch_C(config-if)#` `ipv6 address 2001:db8:3::4/64`	Specify the IPv6 address and mask for interface `vlan3`.
`Switch_C(config-if)#` `exit`	Return to Global Configuration mode.
`Switch_C(config)#` `exit`	Return to Privileged Exec mode.
`Switch_C#` `copy running-config startup-config`	Copy the running-config to the startup-config to enable this configuration to execute after restarting Switch_C.

Commands Used

advertisement-interval
enable (VRRP)
preempt-mode
priority
router ipv6 vrrp (interface)
virtual-ipv6

Validation Commands

show vrrp ipv6

VRRPv3 configuration validation commands and output

Switch_A To display information about the configured VRRPv3 session, to validate configuration as the Master Virtual Router following the earlier configuration steps, enter the command:

```
Switch_A# show vrrp ipv6 vlan2
```

Output **Figure 81-1: Example output from the show vrrp ipv6 vlan2 command on Switch_A**

```
Switch_A#show vrrp ipv6 vlan2
VrId <2>
 State is Master
 Virtual IP is 2001:db8::1 (Not-owner)
 Interface is vlan2
 Priority is 255
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Switch_B To display information about the configured VRRPv3 session, to validate configuration as the Backup Virtual Router following the earlier configuration steps, enter the command:

```
Switch_B# show vrrp ipv6 vlan2
```

Output **Figure 81-2: Example output from the show vrrp ipv6 vlan2 command on Switch_B**

```
Switch_B#show vrrp ipv6 vlan2
VrId <2>
 State is Backup
 Virtual IP is 2001:db8::1 (Not-owner)
 Interface is vlan2
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Then disable the Master Virtual Router and validate so the Backup Virtual Router takes over:

`Switch_A#` `configure terminal`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_A(config)#` `interface vlan2`	Specify the interface (`vlan2`) that will participate in virtual routing to first configure an IP address for `vlan2`.
`Switch_A(config)#` `shutdown`	Shutdown the interface (`vlan2`) that participates as the Master Virtual Router on Switch_A. This will make Switch_B, which is configured as the Backup Virtual Router, become the Master Virtual Router.

Switch_A To validate Switch_A is not the Master Virtual Router after a shutdown on interface vlan2, so Switch_B will take over as Master Virtual Router from Switch_A. To check this, enter the command:

```
Switch_A# show vrrp ipv6 vlan2
```

Output **Figure 81-3: Example output from the show vrrp ipv6 vlan2 command on Switch_A**

```
Switch_A#show vrrp ipv6 vlan2
VrId <2>
 State is Initialize
 Virtual IP is 2001:db8::1 (Owner)
 Interface is vlan2
 Priority is 255
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Switch_B To validate Switch_B is the Master Virtual Router after a shutdown on interface vlan2 on Switch_A, so Switch_B takes over as the Master Virtual Router, enter the command:

```
Switch_B# show vrrp ipv6 vlan2
```

Output **Figure 81-4: Example output from the show vrrp ipv6 vlan2 command on Switch_B**

```
Switch_B#show vrrp ipv6 vlan2
VrId <2>
 State is Master
 Virtual IP is 2001:db8::1 (Not-owner)
 Interface is vlan2
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Chapter 82: VRRP Commands

Command List 82.2
advertisement-interval 82.3
circuit-failover 82.5
debug vrrp 82.7
debug vrrp events 82.8
debug vrrp packet 82.9
disable (VRRP) 82.10
enable (VRRP) 82.11
preempt-mode 82.12
priority 82.14
router vrrp (interface) 82.16
router ipv6 vrrp (interface) 82.17
show debugging vrrp 82.18
show running-config router vrrp 82.18
show running-config router ipv6 vrrp 82.19
show vrrp 82.20
show vrrp ipv6 ..... 82.21
show vrrp counters 82.22
show vrrp (session) 82.24
transition-mode 82.25
undebug vrrp 82.26
undebug vrrp events 82.26
undebug vrrp packet 82.27
virtual-ip 82.28
virtual-ipv6 ..... 82.30
vrrp vmac 82.32

Command List

This chapter provides an alphabetical reference for commands used to configure the Virtual Router Redundancy Protocol (VRRP). For more information, see **Chapter 81, VRRP Introduction and Configuration**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

advertisement-interval

Use this command to configure the advertisement interval of the virtual router. This is the length of time, in seconds, between each advertisement sent from the master to its backup(s).

IPv6 VRRP advertisements are sent to the multicast address assigned to the VRRP group (ff02:0:0:0:0) and a backup virtual router has to join all multicast groups within this range. VRRP advertisements are sent to a multicast address (ff02::12) every second by default.

Use the **no** variant of this command to remove an advertisement interval of the virtual router, which has been set using the **advertisement-interval** command, and revert to the default advertisement interval of 1 second.

Syntax

```
advertisement-interval [<1-255>|csec <1-4095>]
no advertisement-interval
```

Parameter	Description
<1-255>	Specifies the advertisement interval in seconds.
`csec`	Use centiseconds instead of seconds for the advertisement interval.
<1-4095>	Specifies the advertisement interval in centiseconds.

Default The default advertisement interval is 1 second.

Mode Router Configuration

Usage Note when using VRRP with VCStacking, ensure the VRRP advertisement-interval is larger than the VCStacking failover time to avoid VCStacking failovers causing VRRP failovers.

See the **Virtual Router Redundancy Protocol** section in **Chapter 81, VRRP Introduction and Configuration** about setting the advertisement-interval when configuring VRRP.

See the **VRRPv2 to VRRPv3 Transition Configuration** section in **Chapter 81, VRRP Introduction and Configuration** about using seconds for VRRPv2 host compatibility whenever you use **transition-mode** to upgrade or transition from VRRPv2 to VRRPv3.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Note

When using VRRPv3 with VCStacking, ensure that the VRRPv3 advertisement-interval is configured to a longer time than the VCStacking failover time. If the VRRPv3 advertisement-interval is shorter than the VCStacking failover time, then a VRRPv3 failover will also occur whenever a VCStacking failover occurs. Use seconds not centiseconds to ensure interoperability with VRRPv2.

Examples

The example below shows you how to configure the advertisement interval to 6 seconds for the VRRP IPv4 session with VR ID 5 on interface vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# advertisement-interval 6
```

The example below shows you how to reset the advertisement interval to the default of 1 second for the VRRP IPv4 session with VR ID 5 on interface vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# no advertisement-interval
```

The example below shows you how to configure the advertisement interval to 6 seconds for the VRRPv3 IPv6 session with VR ID 5 on interface vlan2:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 5 vlan2
awplus(config-router)# advertisement-interval 6
```

Related Commands

router vrrp (interface)
router ipv6 vrrp (interface)

circuit-failover

Use this command to enable the VRRP circuit failover feature. See the **VRRP Circuit Failover Configuration** in **Chapter 81, VRRP Introduction and Configuration**.

Use the **no** variant of this command to disable this feature.

Syntax

```
circuit-failover <interface> <1-253>
no circuit-failover [<interface> <1-253>]
```

Parameter	Description
<interface>	The interface of the router that is monitored. Interface must exist on the router, and is usually an upstream interface. Should the interface go down, then another router that is configured as a backup router in the group takes over as the master. You should configure the circuit failover on an interface other than the active VRRP interface.
<1-253>	Delta value. The value by which virtual routers decrement their priority value during a circuit failover event. Configure this value to be greater than the difference of priorities on the master and backup routers. In the case of failover, this priority delta value is subtracted from the current VR Master Router priority value.

Mode Router Configuration

Examples The example below shows you how to configure circuit failover on interface vlan2 for the VRRP IPv4 session with VR ID 1, where interface vlan2 is considered the monitored interface:

```
awplus# configure terminal
awplus(config)# router vrrp 1 vlan2
awplus(config-router)# circuit-failover vlan2 30
```

The example below shows you how to remove all configured circuit failovers for the VRRP IPv4 session with VR ID 1 on interface vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 1 vlan2
awplus(config-router)# no circuit-failover
```

The example below shows you how to configure circuit failover on interface vlan2 for the VRRPv3 IPv6 session with VR ID 2, where interface vlan2 is considered the monitored interface:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 2 vlan2
awplus(config-router)# circuit-failover vlan2 30
```

The example below shows you how to remove all configured circuit failovers for the VRRPv3 IPv6 session with VR ID 1 on interface vlan2:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 1 vlan2
awplus(config-router)# no circuit-failover
```

Related Commands **router vrrp (interface)**
router ipv6 vrrp (interface)

debug vrrp

Use this command to specify debugging options for VRRP. The **all** parameter turns on all the debugging options.

Use the **no** variant of this command to disable this function.

Syntax

```
debug vrrp [all]
no debug vrrp [all]
```

Mode Privileged Exec and Global Configuration

Usage For VRRPv3 debugging details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Debugging" on page 81.12**.

Examples The example below shows you how to enable all debugging for VRRP:

```
awplus# configure terminal
awplus(config)# debug vrrp all
```

The example below shows you how to disable all debugging for VRRP:

```
awplus# configure terminal
awplus(config)# no debug vrrp all
```

Related Commands
show debugging vrrp
undebug vrrp

debug vrrp events

Use this command to specify debugging options for VRRP event troubleshooting.

Use the **no** variant of this command to disable this function.

Syntax

```
debug vrrp events
no debug vrrp events
```

Mode Privileged Exec and Global Configuration

Usage The **debug vrrp events** command enables the display of debug information related to VRRP internal events.

For VRRPv3 debugging details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Debugging" on page 81.12**.

Examples The example below shows you how to enable events debugging for VRRP:

```
awplus# configure terminal
awplus(config)# debug vrrp events
```

The example below shows you how to disable events debugging for VRRP:

```
awplus# configure terminal
awplus(config)# no debug vrrp events
```

Related Commands

show debugging vrrp
undebug vrrp events

debug vrrp packet

Use this command to specify debugging options for VRRP packets.

Use the **no** variant of this command to disable this function.

Syntax

```
debug vrrp packet [send|recv]
no debug vrrp packet [send|recv]
```

Parameter	Description
`send`	Specifies the debug option set for sent packets.
`recv`	Specifies the debug option set for received packets.

Mode Privileged Exec and Global Configuration

Usage The **debug vrrp packet** command enables the display of debug information related to the sending and receiving of packets.

For VRRPv3 debugging details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Debugging" on page 81.12**.

Examples The example below shows you how to enable received and sent packet debugging for VRRP:

```
awplus# configure terminal
awplus(config)# debug vrrp packet
```

The example below shows you how to enable only received packet debugging for VRRP:

```
awplus# configure terminal
awplus(config)# debug vrrp packet recv
```

The example below shows you how to enable only sent packet debugging for VRRP:

```
awplus# configure terminal
awplus(config)# debug vrrp packet send
```

The example below shows you how to disable packet debugging for VRRP:

```
awplus# configure terminal
awplus(config)# no debug vrrp packet
```

Related Commands **show debugging vrrp**
undebug vrrp packet

disable (VRRP)

Use this command to disable a VRRP IPv4 session or a VRRPv3 IPv6 session on the router to stop it participating in virtual routing. Note that when this command is configured then a backup router assumes the role of master router depending on its priority. See the **enable (VRRP)** command to enable a VRRP IPv4 session or a VRRPv3 IPv6 session on the router.

Syntax

```
disable
```

Mode

Router Configuration

Usage

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples

The example below shows you how to disable the VRRP session for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# disable
```

The example below shows you how to disable the VRRPv3 session for VRRPv3 VR ID 3 on vlan1:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# disable
```

Related Commands

enable (VRRP)
router vrrp (interface)
router ipv6 vrrp (interface)
show vrrp

enable (VRRP)

Use this command to enable the VRRP session on the router to make it participate in virtual routing. To make an changes to the VRRP configuration, first disable the router from participating in virtual routing using the **disable (VRRP)** command.

Syntax `enable`

Mode Router Configuration

Usage You must configure the virtual IP address and define the interface for the VRRP session (using the **virtual-ip** or **virtual-ipv6** and the **router vrrp (interface)** or **router ipv6 vrrp (interface)** commands) before using this command.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples The example below shows you how to enable the VRRP session for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# enable
```

The example below shows you how to enable the VRRPv3 session for VRRPv3 VR ID 3 on vlan1:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# enable
```

Related Commands
disable (VRRP)
router vrrp (interface))
router ipv6 vrrp (interface)
show vrrp
virtual-ip
virtual-ipv6

preempt-mode

Use this command to configure preempt mode. If preempt-mode is set to **true**, then the highest priority backup will always be the master when the default master is unavailable.

If preempt-mode is set to **false**, then a higher priority backup will not preempt a lower priority backup who is acting as master.

Syntax `preempt-mode {true|false}`

Parameter	Description
`true`	Preemption is enabled.
`false`	Preemption is disabled.

Default The default is **true**.

Mode Router Configuration

Usage When the master router fails, the backup routers come online in priority order—highest to lowest. Preempt mode means that a higher priority back up router will take over the master role from a lower priority back up. Preempt mode on **true** allows a higher priority backup router to relieve a lower priority backup router.

By default, a preemptive scheme is enabled whereby a higher priority backup virtual router that becomes available take over for the backup virtual router that was elected to become the master virtual router.

This preemptive scheme can be disabled using the **preempt-mode false** command. If preemption is disabled, the backup virtual router that is currently elected as the master virtual router does not transition to backup virtual router again whenever the alternate backup router with a higher priority becomes available.

See **"VRRP Election and Preempt for IPv6" on page 81.10** for further information on preempt mode.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples The example below shows you how to configure preempt-mode as true for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# preempt-mode true
```

The example below shows you how to configure preempt-mode as false for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# preempt-mode false
```

The example below shows you how to configure preempt-mode as true for VRRPv3 VR ID 3 on vlan1:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# preempt-mode true
```

The example below shows you how to configure preempt-mode as false for VRRPv3 VR ID 3 on vlan1:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# preempt-mode false
```

Related Commands

circuit-failover
priority
router vrrp (interface)
router ipv6 vrrp (interface)

priority

Use this command to configure the VRRP router priority within the virtual router. The highest priority router is Master (unless **preempt-mode** is false).

Use the **no** variant of this command to remove the VRRP router priority within the virtual router, which has been set using the **priority** command.

Syntax

```
priority <1-255>
no priority
```

Parameter	Description
<1-255>	The priority. For the master router, use 255 for this parameter; otherwise use any number from the range <1-254>.

Default Defaults for priority are: **master router** = 255; **backup** = 100.

Mode Router Configuration

Usage Priority determines the role that each VRRP router plays and what happens if the master virtual router fails. If a VRRP router owns the IP address of the virtual router and the IP address of the interface, then this VRRP router functions as the master virtual router.

Priority also determines whether a VRRP router functions as a backup virtual router and the order of ascendancy to becoming a master virtual router if the master virtual router fails. Configure the priority of each backup virtual router with a a value of 1 through 254.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples The example below shows you how to configure 101 as the priority for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# priority 101
```

The example below shows you how to remove the priority configured for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# no priority
```

The example below shows you how to configure 101 as the priority for VRRPv3 VR ID 3 on vlan1:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# priority 101
```

The example below shows you how to remove the configured priority for VRRPv3 VR ID 3 on vlan1:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# no priority
```

Related Commands **circuit-failover**
preempt-mode

router vrrp (interface)

Use this command to configure VRRP IPv4 and define the interface that will participate in virtual routing to send and receive advertisement messages. This command allows you to enter the Router Configuration mode.

Use the **no** variant of this command to remove the VRRP IPv4 configuration. Disable the VRRP session before using the **no** variant of this command.

Syntax

```
router vrrp <vrid> <interface>
no router vrrp <vrid> <interface>
```

Parameter	Description
<vrid>	*<1-255>* The ID of the virtual router VRRP IPv4 session to create.
<interface>	Specify the name of the interface that will participate in the virtual routing. The interface must exist on the router. The interface specified sends and receives VRRP IPv4 advertisement messages.

Mode Global Configuration

Usage Use the required *<interface>* placeholder to define the interface that will participate in virtual routing. This interface is used for two purposes - to send/receive advertisement messages and to forward on behalf of the virtual router when in master state.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

Examples The example below shows you how to enable a VRRP session with VR ID 5 on vlan1:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan1
awplus(config-router)# enable
awplus(config-router)#
```

The example below shows you how to disable a VRRP session with VR ID 5 on vlan1:

```
awplus(config-router)# disable
awplus(config-router)# exit
awplus(config)# no router vrrp 5 vlan1
awplus(config)#
```

Related Commands
advertisement-interval
circuit-failover
disable (VRRP)
enable (VRRP)

router ipv6 vrrp (interface)

Use this command to configure VRRPv3 for IPv6 and define the interface that will participate in virtual routing to send and receive advertisement messages. This command allows you to enter the Router Configuration mode.

Use the **no** variant of this command to remove the VRRPv3 for IPv6 configuration. Disable the VRRP session before using the **no** variant of this command.

Syntax

```
router ipv6 vrrp <vrid> <interface>
no router ipv6 vrrp <vrid> <interface>
```

Parameter	Description
<vrid>	*<1-255>* The ID of the virtual router VRRPv3 IPv6 session to create.
<interface>	Specify the name of the interface that will participate in the virtual routing. The interface must exist on the router. The interface specified sends and receives VRRPv3 IPv6 advertisement messages.

Mode Global Configuration

Usage Use the required *<interface>* placeholder to define the interface that will participate in virtual routing. This interface is used for two purposes - to send/receive advertisement messages and to forward on behalf of the virtual router when in master state.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples The example below shows you how to enable a VRRPv3 session with VR ID 3 on vlan2:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan2
awplus(config-router)# enable
awplus(config-router)#
```

The example below shows you how to disable a VRRPv3 session with VR ID 3 on vlan2:

```
awplus(config-router)# disable
awplus(config-router)# exit
awplus(config)# no router ipv6 vrrp 3 vlan2
awplus(config)#
```

Related Commands **advertisement-interval**
circuit-failover

show debugging vrrp

Use this command to display the set VRRP debugging option. Use the **terminal monitor** command to display output on the console otherwise debug output is in the log file.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 debugging details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Debugging" on page 81.12**.

Syntax
```
show debugging vrrp
```

Mode User Exec and Privileged Exec

Example The example below shows you how to display VRRP debugging:

```
awplus# show debugging vrrp
```

Related Commands
debug vrrp
debug vrrp events
debug vrrp packet

show running-config router vrrp

Use this command to show the running configuration for VRRP IPv4.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

Syntax
```
show running-config router vrrp
```

Mode Privileged Exec, Global Configuration, Line Configuration, and Interface Configuration.

Example The example below shows you how to display the running configuration for VRRP IPv4:

```
awplus# show running-config router vrrp
```

Output **Figure 82-1: Example output from the show running-config router vrrp command**

```
!
router vrrp 2 vlan2
 circuit-failover vlan2 2
 advertisement-interval 4
 preempt-mode true
!
```

show running-config router ipv6 vrrp

Use this command to show the running configuration for VRRPv3 IPv6.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Syntax `show running-config router vrrp`

Mode Privileged Exec, Global Configuration, Line Configuration, and Interface Configuration.

Example The example below shows you how to display the running configuration for VRRPv3 IPv6:

```
awplus# show running-config router ipv6 vrrp
```

Output **Figure 82-2: Example output from the show running-config router ipv6 vrrp command**

```
!
router ipv6 vrrp 3 vlan3
 virtual-ip fe80::202:b3ff:fed5:983e master
 circuit-failover vlan3 3
 advertisement-interval 6
 preempt-mode false
!
```

show vrrp

Use this command to display information about all VRRP IPv4 sessions. This command shows a summary when the optional **brief** parameter is used.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

Syntax

```
show vrrp [brief]
```

Parameter	Description
`brief`	Brief summary of VRRP sessions.

Mode User Exec and Privileged Exec

Example To display information about all VRRP IPv4 sessions, enter the command:

```
awplus# show vrrp
```

To display brief summary output about VRRP IPv4 sessions, enter the command:

```
awplus# show vrrp brief
```

Output

Figure 82-3: Example output from the show vrrp command

```
awplus#show vrrp
VMAC enabled
Address family IPv4
VRRP Id: 1 on interface: vlan2
 State: AdminUp   - Master
 Virtual IP address: 192.168.1.2 (Not-owner)
 Priority is 100
 Advertisement interval: 100 centiseconds
 Preempt mode: TRUE
 Multicast membership on IPv4 interface vlan2: JOINED
 Transition mode: FALSE
 Accept mode: FALSE
 Master address: 192.168.1.3
```

Figure 82-4: Example output from the show vrrp brief command

```
awplus#show vrrp brief
Interface        Grp  Prio Own  Pre  State      Master addr      Group addr
 vlan10            1   200   N    P  Master     192.168.10.4     192.168.10.253
 vlan10            2   150   N    P  Backup     192.168.10.4     192.168.10.254
 vlan11            3   200   N    P  Master     192.168.11.4     192.168.11.253
 vlan11            4   150   N    P  Backup     192.168.11.4     192.168.11.254
```

Related Commands

enable (VRRP)
disable (VRRP)

show vrrp ipv6

Use this command to display information about all configured VRRPv3 IPv6 sessions for all interfaces, or all VRRPv3 IPv6 sessions for a given interface with the optional parameter.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Syntax

```
show vrrp ipv6 [<interface>]
```

Parameter	Description
<interface>	Specify the name of the interface that will participate in the virtual routing. The interface must exist on the router. The interface specified sends and receives VRRPv3 IPv6 advertisement messages.

Mode User Exec and Privileged Exec

Example To display information about all VRRPv3 IPv6 sessions, enter the command:

```
awplus# show vrrp ipv6
```

Output **Figure 82-5: Example output from the show vrrp ipv6 vlan2 command**

```
awplus#show vrrp ipv6 vlan2
VrId <1>
 State is Master
 Virtual IP is fe80::202:b3ff:fed5:983e (Owner)
 Interface is vlan2
 Priority is 255
 Advertisement interval is 4 sec
 Preempt mode is FALSE
```

Related Commands

enable (VRRP)
disable (VRRP)

show vrrp counters

This command displays VRRP SNMP counters on the console, as described in the VRRP MIB and RFC2787, for debugging use while you configure VRRP with commands in this chapter.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

Syntax `show vrrp counters`

Mode User Exec and Privileged Exec

Usage The output has a section for global counters and a section of counters for each VRRP instance configured. See the descriptions of the counters below the sample output as per RFC2787.

Note

Note that the counters displayed with this commands are the same counters as described in RFC 2787 (Copyright (C) The Internet Society (2000). All Rights Reserved) except for the `Monitored Circuit Up` and `Monitored Circuit Down` counters which are additions beyond the MIB.

Example To display information about VRRP SNMP counters on the console, enter the command:

```
awplus# show vrrp counters
```

Figure 82-6: Example output from the show vrrp counters command

```
awplus#show vrrp counters
VRRP Global Counters:
  Checksum Errors .... 230
  Version Errors ..... 0
  VRID Errors ........ 230

VRRP IPv4 counters for VR 10/vlan10:
  Master Transitions ........ 0
  Received Advertisements ... 0
  Internal Errors ........... 0
  TTL Errors ................ 0
  Received Priority 0 Pkt ... 0
  Sent Priority 0 Pkt ....... 0
  Received Invalid Type ..... 0
  Address List Errors ....... 0
  Packet Length Errors ...... 0
  Monitored Circuit Up ...... 0
  Monitored Circuit Down..... 0

VRRP IPv4 counters for VR 100/vlan100:
  Master Transitions ........ 1
  Received Advertisements ... 1614
  Internal Errors ........... 0
  TTL Errors ................ 0
  Received Priority 0 Pkt ... 0
  Sent Priority 0 Pkt ....... 0
  Received Invalid Type ..... 0
  Address List Errors ....... 0
  Packet Length Errors ...... 0
  Monitored Circuit Up ...... 0
  Monitored Circuit Down..... 2
```

Table 82-1: Global counters with descriptions for the show vrrp counters command:

Counter	Description
`Checksum Errors`	The total number of VRRP packets received with an invalid VRRP checksum value.
`Version Errors`	The total number of VRRP packets received with an unknown or unsupported version number.
`VRID Errors`	The total number of VRRP packets received with an invalid VRID for this virtual router.

Table 82-2: Per VR counters with descriptions for the show vrrp counters command:

Counter	Description
`Master Transitions`	The total number of times that this virtual router's state has transitioned to MASTER.
`Received Advertisements`	The total number of VRRP advertisements received by this virtual router.
`Internal Errors`	The total number of VRRP advertisement packets received for which the advertisement interval is different than the one configured for the local virtual router.
`TTL Errors`	The total number of VRRP packets received by the virtual router with IP TTL (Time-To-Live) not equal to 255.
`Received Priority 0 Pkt`	The total number of VRRP packets received by the virtual router with a priority of '0'.
`Sent Priority 0 Pkt`	The total number of VRRP packets sent by the virtual router with a priority of '0'.
`Received Invalid Type`	The number of VRRP packets received by the virtual router with an invalid value in the 'type' field.
`Address List Errors`	The total number of packets received for which the address list does not match the locally configured list for the virtual router.
`Packet Length Errors`	The total number of packets received with a packet length less than the length of the VRRP header.
`Monitored Circuit Up`	The total number of times the monitored circuit has generated the UP event.
`Monitored Circuit Down`	The total number of times the monitored circuit has generated the down event.

show vrrp (session)

Use this command to display information for a particular VRRP session.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

Syntax `show vrrp <vrid> <interface>`

Parameter	Description
<vrid>	*<1-255>* The virtual router ID for which to display information. Session must already exist.
<interface>	The interface to display information about, for instance, `vlan2`.

Mode User Exec and Privileged Exec

Usage See the below sample output from the **show vrrp** command displaying information about VRRP session 1 configured on **vlan2**. Output shows that a Virtual IP address has been set.

```
awplus# show vrrp 1 vlan2
```

```
awplus#show vrrp 1 vlan2
Address family IPv4
VrId <1>
 Interface is vlan2
 State is Initialize
 Virtual IP address is 10.10.11.250 (Not IP owner)
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

See the below sample output from the **show vrrp** command displaying information about VRRP session 1 configured on **vlan3**. Output shows a Virtual IP address has not been set.

```
awplus# show vrrp 1 vlan3
```

```
awplus#show vrrp 1 vlan3
Address family IPv4
VrId <1>
 Interface is vlan3
 State is Initialize
 Virtual IP address is unset
 Priority is 100
 Advertisement interval is 1 sec
 Preempt mode is TRUE
```

Example The following command shows information about VRRP session 5 for interface **vlan2**.

```
awplus# show vrrp 5 vlan2
```

transition-mode

Use this command to configure the IPv4 transition mode. Transition mode allows you to upgrade from VRRPv2 to VRRPv3 and gives interoperability between VRRPv2 and VRRPv3.

If transition-mode is set to **true**, then the IPv4 transition mode is enabled and VRRPv2 and VRRPv3 advertisements are sent allowing VRRPv2 and VRRPv3 interoperability. Received VRRPv2 advertisement packets are accepted and processed when transition-mode is true.

If transition-mode is set to **false**, then the IPv4 transition mode is disabled and only VRRPv3 advertisements are sent. Received VRRPv2 advertisement packets are dropped.

Note the **advertisement-interval** should not be configured to less than 1 second when using transition-mode. VRRPv2 can only use advertisements in whole second intervals.

Syntax `transition-mode {true|false}`

Parameter	Description
`true`	Transition mode is enabled. This results in VRRPv2 and VRRPv3 IPv4 advertisements being sent. Transition mode is only available on VRRPv3 for interoperability with VRRPv2 while upgrading to VRRPv3.
`false`	Transition mode is disabled. This stops VRRPv2 IPv4 advertisements being sent. Only VRRPv3 advertisements are sent when disabled. Disable transition-mode after upgrading from VRRPv2 to VRRPv3.

Default The default is **false**.

Mode Router Configuration

Usage See **"VRRPv2 to VRRPv3 Transition Configuration" on page 81.23** for further information about configuring transition mode to upgrade from VRRPv2 to VRRPv3.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples The example below shows you how to configure IPv4 transition-mode as true for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# transition-mode true
```

The example below shows you how to configure IPv4 transition-mode as false for VRRP VR ID 5 on vlan2:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# transition-mode false
```

Related Commands router vrrp (interface)

undebug vrrp

Use this command to disable all VRRP debugging.

Syntax

```
undebug vrrp all
```

Mode Privileged Exec

Example The example below shows you how to disable all VRRP debugging:

```
awplus# undebug vrrp all
```

Related Commands debug vrrp

undebug vrrp events

Use this command to disable debugging options for VRRP event troubleshooting.

Syntax

```
undebug vrrp events
```

Mode Privileged Exec

Example The example below shows you how to disable VRRP event debugging:

```
awplus# undebug vrrp events
```

Related Commands debug vrrp events

undebug vrrp packet

Use this command to disable debugging options for VRRP packets.

Syntax `undebug vrrp packet [send|recv]`

Parameter	Description
`send`	Disable the debug option set for sent packets.
`recv`	Disable the debug option set for received packets.

Mode Privileged Exec

Examples The example below shows you how to disable VRRP sent packet debugging:

```
awplus# undebug vrrp packet send
```

The example below shows you how to disable VRRP received packet debugging:

```
awplus# undebug vrrp packet recv
```

The example below shows you how to disable all VRRP packet debugging:

```
awplus# undebug vrrp packet
```

Related Commands **debug vrrp packet**

virtual-ip

Use this command to set the virtual IP address for the VRRP session. This is the IP address of the virtual router that end hosts set as their default gateway.

Use the **no** variant of this command to disable this feature.

Syntax

```
virtual-ip <ip-address> [master|backup|owner]
no virtual-ip
```

Parameter	Description
<ip-address>	The virtual IPv4 address of the virtual router, entered in dotted decimal format A.B.C.D.
`master`	Sets the default state of the VRRP router within the Virtual Router as **master**. For master, the router must own the Virtual IP address. Specify the **owner** option before using **master** option.
`backup`	Sets the default state of the VRRP router within the Virtual Router as **backup**.
`owner`	Sets the IPv6 address of the VRRP router within the Virtual Router as the **owner**. Specify this before using the **master** option.

Mode Router Configuration

Usage The VRRP master and owner of the virtual IPv4 address for the VRRP session only responds to the packets destined to the virtual IPv6 address. The VRRP master that is not an owner of the virtual IPv4 address for the VRRP session does not respond to the packets destined to the virtual IPv4 address, but forwards packets with a VMAC as the destination address. See the **vrrp vmac** command to enable and disable this feature.

For VRRPv3 IPv4 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP Configuration for IPv4" on page 81.4**.

Examples The example below shows you how to set the virtual IP address for VRRP VR ID 5 and the router as the VRRP master:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# virtual-ip 192.0.2.30 master
```

The example below shows you how to set the virtual IPv4 address for VRRP VR ID 5 and the router as the VRRP backup:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# virtual-ip 192.0.2.30 backup
```

The example below shows you how to set the virtual IPv4 address for VRRP VR ID 5 and the router as owner of the virtual IPv4 address:

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# virtual-ip 192.0.2.30 owner
```

The example below shows you how to disable the virtual IPv4 address for VRRP VR ID 5

```
awplus# configure terminal
awplus(config)# router vrrp 5 vlan2
awplus(config-router)# no virtual-ip
```

Related Commands

router vrrp (interface)
enable (VRRP)
vrrp vmac

virtual-ipv6

Use this command to set the virtual IPv6 address for the VRRPv3 session. This is the IPv6 address of the virtual router that end hosts set as their default gateway.

Note that the IPv6 address specified is an IPv6 link-local address. See the **Usage** note below for further information.

Use the **no** variant of this command to disable this feature.

Syntax

```
virtual-ipv6 <ipv6-address> [master|backup|owner]
no virtual-ipv6
```

Parameter	Description
<ipv6-address>	The virtual IPv6 address of the virtual router, entered in hexadecimal, in the format X:X::X.X. This is an IPv6 link-local address. See the **Usage** note below for further information.
`master`	Sets the default state of the VRRPv3 router within the Virtual Router as **master**. For master, the router must own the Virtual IP address. Specify the **owner** option before using **master** option.
`backup`	Sets the default state of the VRRPv3 router within the Virtual Router as **backup**.
`owner`	Sets the IPv6 address of the VRRPv3 router within the Virtual Router as the **owner**. Specify this before using the **master** option.

Mode Router Configuration

Usage The VRRP master and owner of the virtual IPv6 address for the VRRPv3 session only responds to the packets destined to the virtual IPv6 address. The VRRP master that is not an owner of the virtual IPv6 address for the VRRPv3 session does not respond to the packets destined to the virtual IPv6 address, but forwards packets with a VMAC as the destination address. See the **vrrp vmac** command to enable and disable this feature.

The AlliedWare Plus VRRPv3 implementation supports one IPv6 virtual link local address per virtual router ID. Note that in the command examples fe80::1 is an IPv6 link-local address. An IPv6 link-local address is used because IPv6 link-local addresses are used by IPv6 ND (Neighbor Discovery). A host's default route to a router points to the IPv6 link-local address, not a specific global IPv6 address for the router. For the host's traffic to switch over to a backup router, the IPv6 link-local address of the router is used by VRRPv3.

For VRRPv3 IPv6 configuration details, see **Chapter 81, VRRP Introduction and Configuration** and the section **"VRRP IPv6 Configuration Example" on page 81.29**.

Examples The example below shows you how to set the virtual IPv6 address for VRRPv3 VR ID 3 and the router as the VRRPv3 master:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# virtual-ipv6 fe80::1 master
```

The example below shows you how to set the virtual IPv6 address for VRRPv3 VR ID 3 and the router as the VRRPv3 backup:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# virtual-ipv6 fe80::1 backup
```

The example below shows you disable the virtual IPv6 address for VRRPv3 VR ID 3:

```
awplus# configure terminal
awplus(config)# router ipv6 vrrp 3 vlan1
awplus(config-router)# no virtual-ipv6
```

Related Commands
router ipv6 vrrp (interface)
enable (VRRP)
vrrp vmac

vrrp vmac

Use this command to enable or disable the VRRP Virtual MAC feature. This feature is used by VRRP to make the hosts use the virtual MAC address as the physical hardware address of their gateway.

A VRRP router master will use the virtual MAC address for any ARP responses associated with the virtual IP address, or any gratuitous ARPs sent on behalf of the virtual IP address.

All VRRP advertisements are sent using this virtual MAC address as the source MAC address.

The virtual MAC address has the form: 00:00:5e:00:01:<VRID>, where VRID is the ID of the Virtual Router.

Syntax `vrrp vmac {enable|disable}`

Mode Global Configuration

Examples To enable Virtual MAC enter:

```
awplus# configure terminal
awplus(config)# vrrp vmac enable
```

To disable Virtual MAC enter:

```
awplus# configure terminal
awplus(config)# vrrp vmac disable
```

Related Commands **virtual-ip**
virtual-ipv6

Chapter 83: EPSR Introduction and Configuration

Introduction 83.2
Ring Components and Operation 83.2
Fault Detection and Recovery 83.4
  Fault Recovery 83.4
  Restoring Normal Operation 83.5
  Managing Rings with Two Breaks 83.6
  Recovery When One Break is Restored 83.8
Configuration Examples 83.10
  Single Domain, Single Ring Network 83.10
  Single Ring, Dual Domain Network 83.16
Interconnected Rings 83.17
  Superloop Protection 83.18
  EPSR Superloop Prevention 83.19
  Configuring a Basic Superloop Protected Two Ring EPSR Network 83.22
  Sample Show Output 83.38
  Adding a New Data VLAN to a Functioning Superloop Topology 83.42
  EPSR and Spanning Tree Operation 83.45

Introduction

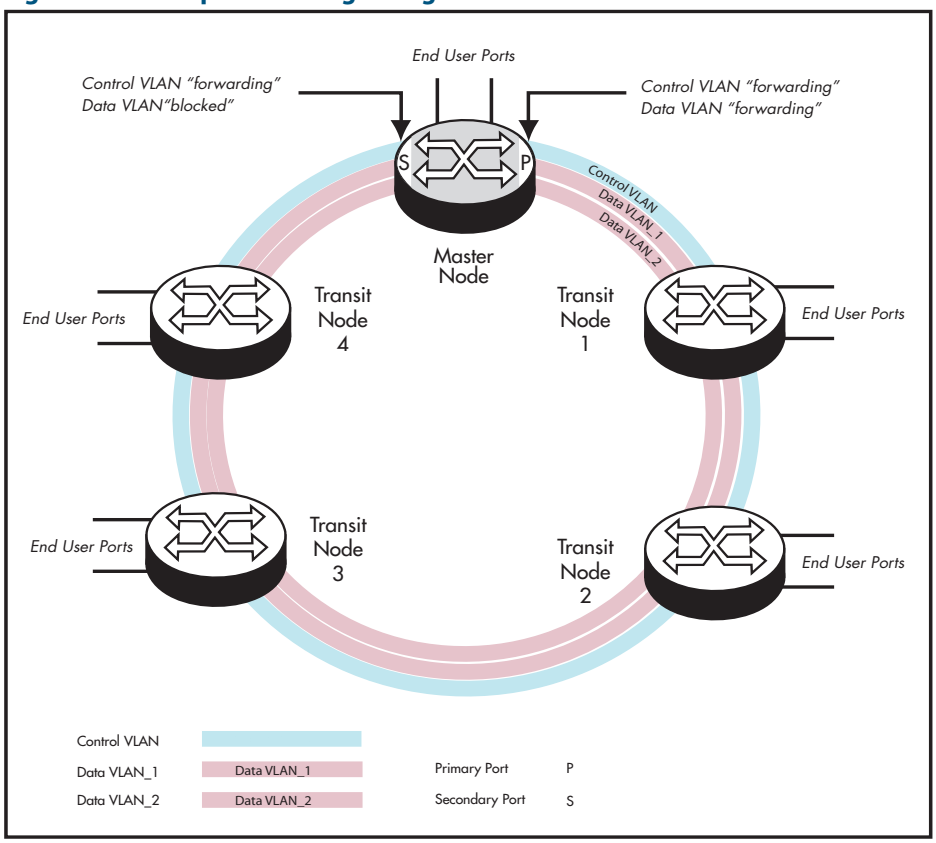
Ethernet Protection Switching Ring (EPSR) is a protection system that prevents loops within Ethernet ring based topologies. EPSR offers a rapid detection and recovery time (in the order of 50 ms, depending on configuration) if a link or node fails. This rapid recovery time makes EPSR a more effective alternative to spanning tree options when using ring-based topologies to create high speed resilient Layer 2 networks.

Ring Components and Operation

EPSR operates only on ring-based topologies. An EPSR ring comprises a series of nodes (Ethernet bridges) connected end to end. The figure below shows a basic ring configuration. A ring comprises one master node and a number of transit nodes. Each node connects to the ring via two ports. On the master node one port is configured to be the primary port and the other, the secondary port.

Note If your switch is operating with a base user license it will function only as a transit node. To run your switch as a master node you will need to purchase a Premium License. On x510L Series switches premium licenses will be available from version 5.4.5 onwards.

Figure 83-1: Simple EPSR ring configuration



EPSR instances and domains

Each physical EPSR ring contains one or more EPSR domains. An EPSR instance can be thought of as a component of an EPSR ring domain that exists on a single node. A set of instances across the whole ring is called a "domain." Therefore a ring whose individual nodes each have two instances results in a two domain ring. Each instance contains a control VLAN and a number of data VLANs.

The EPSR control VLAN and its associated data VLANs form a Ring Domain. Although a physical ring can have more than one domain, each domain must operate as a separate logical group of VLANs and must have its own master node. This means that several domains may share the same physical network, but must operate as logically separate VLAN groups.

Control VLAN The function of the control VLAN is to monitor the ring domain and maintain its operational functions. To do this it transmits and monitors operational healthcheck messages using EPSR healthcheck control frames. The control VLAN carries no user data.

Data VLAN The data VLAN carries the user data around the ring. Several data VLANs can share a common control VLAN.

Master node The master node controls the ring operation. It issues healthcheck messages at regular intervals from its primary port and monitors their arrival back at its secondary port - after they have circled the ring. Under normal operating conditions the master node's secondary port is always in the blocking state to all data VLAN traffic. This is to prevent data loops forming within the ring. This port however, operates in the forwarding state for the traffic on the control VLAN. Loops do not occur on the control VLAN because the control messages stop at the secondary port, having completed their path around the ring.

Note If your switch is operating with a base user license it will function only as a transit node. To run your switch as a master node you will need to purchase a Premium License.

Transit nodes The transit nodes operate as conventional Ethernet bridges, but with the additional capability of running the EPSR protocol. This protocol requires the transit nodes to forward the healthcheck messages from the master node, and respond appropriately when a ring fault is detected. The fault condition procedure is explained in **"Fault Detection and Recovery" on page 83.4**.

Fault Detection and Recovery

EPSR uses the following methods to detect outages in a node or a link in the ring:

- Master node polling fault detection
- Transit node unsolicited fault detection

Master node polling

The master node issues healthcheck messages from its primary port as a means of checking the condition of the EPSR network ring. These messages are sent at regular periods, controlled by the **hellotime** parameter of the **epsr** command on page 84.4. A failover timer is set each time a healthcheck message leaves the master node's primary port. The timeout value for this timer is set by the **failover** parameter of the **epsr** command on page 84.4. If the failover timer expires before the transmitted healthcheck message is received by the master node's secondary port, the master node assumes that there is a fault in the ring, and implements its fault recovery procedures. Because this method relies on a timer expiry, its operation is inherently slower than the "transit node unsolicited detection method" described next.

Transit node unsolicited

Transit node unsolicited fault detection relies on transit nodes detecting faults at their interfaces, and immediately notifying master nodes about the break. When a transit node detects a connectivity loss, it sends a "links down" message over its good link. Because a link spans two nodes, both nodes send the "links down" message back to the master node. These nodes also change their state from "links up" to "links down," and change the state of the port connecting to the broken link, from "forwarding" to "blocking."

Fault Recovery

When the master node detects an outage in the ring by using its detection methods, it does the following:

1. Declares the ring to be in a "failed" state.
2. Unblocks its secondary port to enable the data VLAN traffic to pass between its primary and secondary ports.
3. Flushes its own forwarding database (FDB) for (only) the two ring ports.
4. Sends an EPSR Ring-Down-Flush-FDB control message to all the transit nodes, via both its primary and secondary ports.

Transit nodes respond to the Ring-Down-Flush-FDB message by flushing their forward databases for each of their ring ports. As the data starts to flow in the ring's new configuration, each of the nodes (master and transit) re-learn their Layer 2 addresses. During this period, the master node continues to send health check messages over the control VLAN. This situation continues until the faulty link or node is repaired. For a multi-domain ring, this process occurs separately for each domain within the ring.

Note

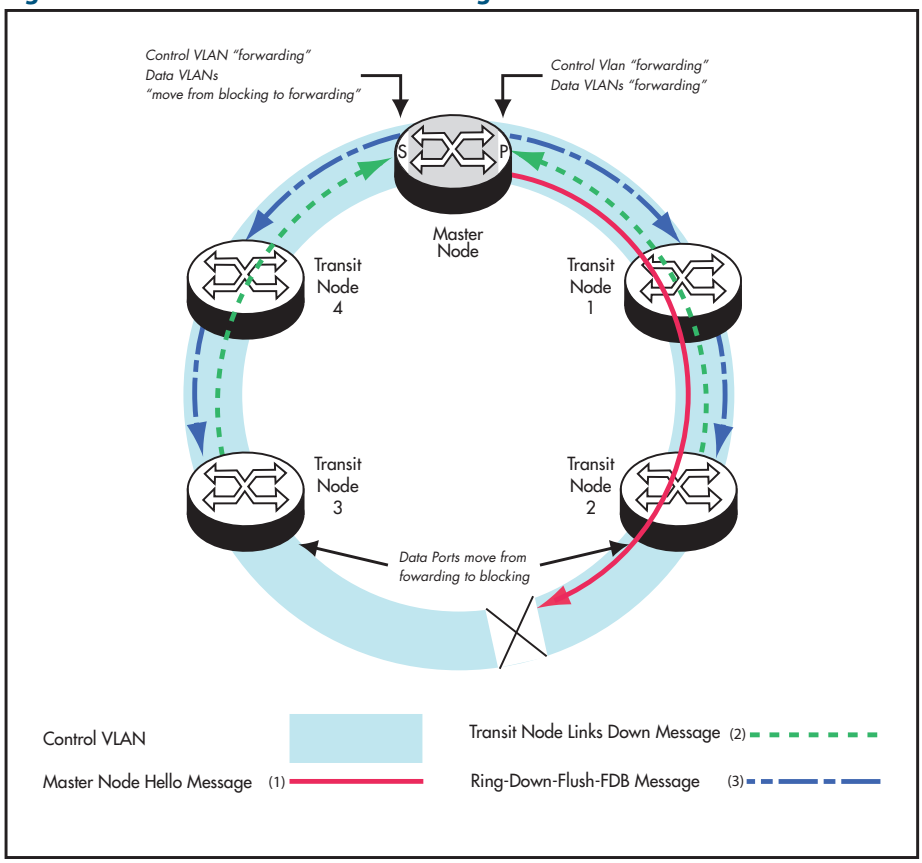
When VCStack is used with EPSR, the EPSR **failovertime** must be set to at least 5 seconds to avoid any broadcast storms during failover. Broadcast storms may occur if the switch cannot failover quickly enough before the EPSR **failovertime** expires.

See the **epsr** command for further information about the EPSR **failovertime**.

See the **reboot rolling** command for further information about VCStack failover.

The following figure shows the flow of control frames under fault conditions.

Figure 83-2: EPSR fault detection messages



Restoring Normal Operation

Transit nodes

Once a fault in the ring or node has been rectified, the transit nodes that span the previously faulty link section detect that link connectivity has returned. They then move their appropriate ring port state, from Links-Down to Pre-Forwarding, and await the Ring-Up-Flush control message from the master node.

Once these transit nodes receive the Ring-Up-Flush message, they:

- flush their forward databases for both their ring ports.
- change the state of their ports from blocking to forwarding, which allows data to flow through their previously blocked ring ports.

Note

The transit nodes do not enter the forward state until they have received the Ring-Up-Flush message. This prevents the possibility of a loop condition occurring caused by the transit nodes moving into the forwarding state before the master node secondary port can return to the blocking state. During such a period, the ring would have no ports blocked.

Master node

With the link restored, the healthcheck messages that are sent from the primary port of the master node now complete the loop and arrive at the master node's secondary port. The master node restores normal conditions as follows:

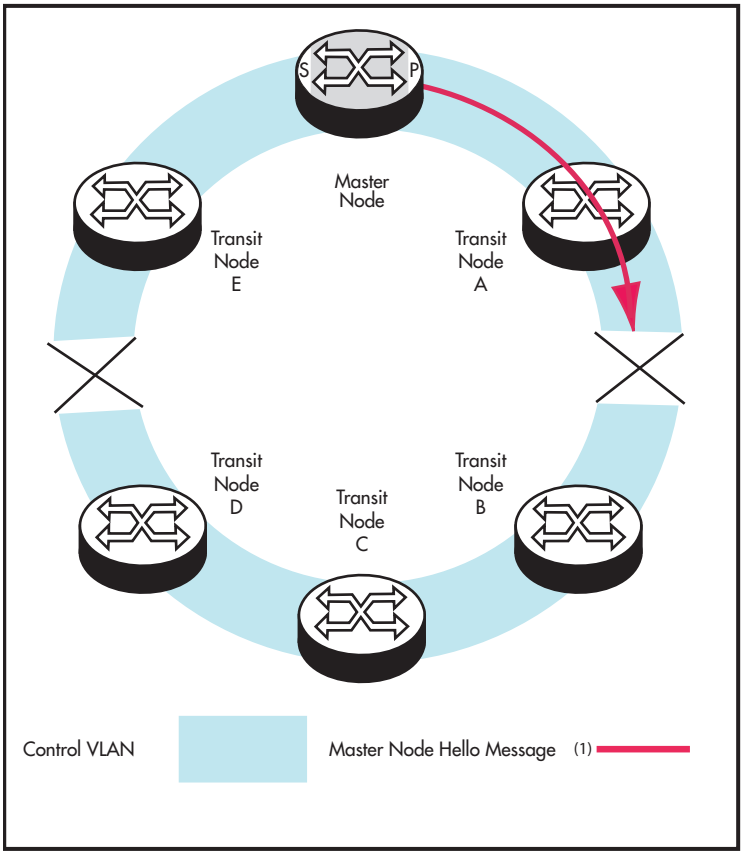
1. Declares the ring to be in a "complete" state.
2. Blocks its secondary port for data (non-control) traffic.
3. Flushes its forwarding database for its two ring ports.

4. Sends a Ring-Up-Flush-FDB message from its primary port, to all transit nodes.

Managing Rings with Two Breaks

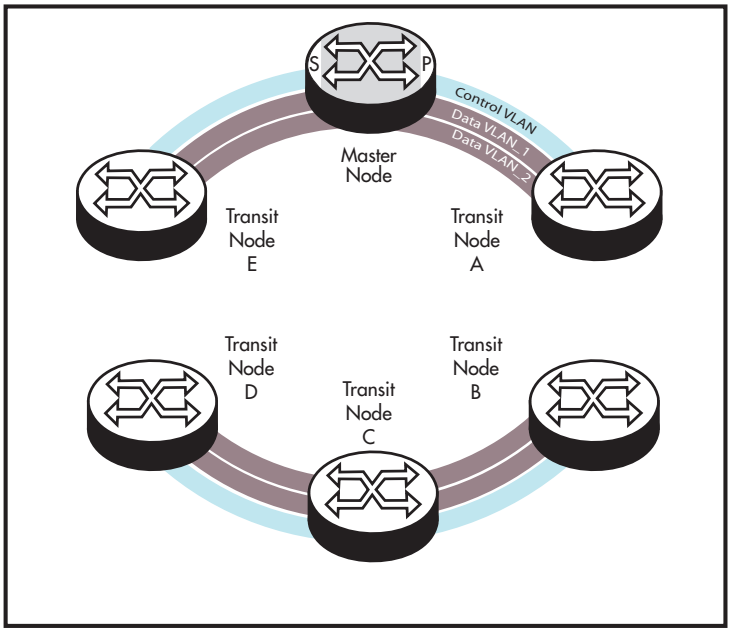
To restore a link with two breaks you need to run the EPSR Enhanced Recovery feature. Consider the network shown below:

Figure 83-3: EPSR ring with two breaks



In this situation the ring will attempt to recover as previously described in **"Fault Recovery" on page 83.4**. This will result in the split-ring operation shown in **Figure 83-4 on page 83.7**.

Figure 83-4: EPSR split ring

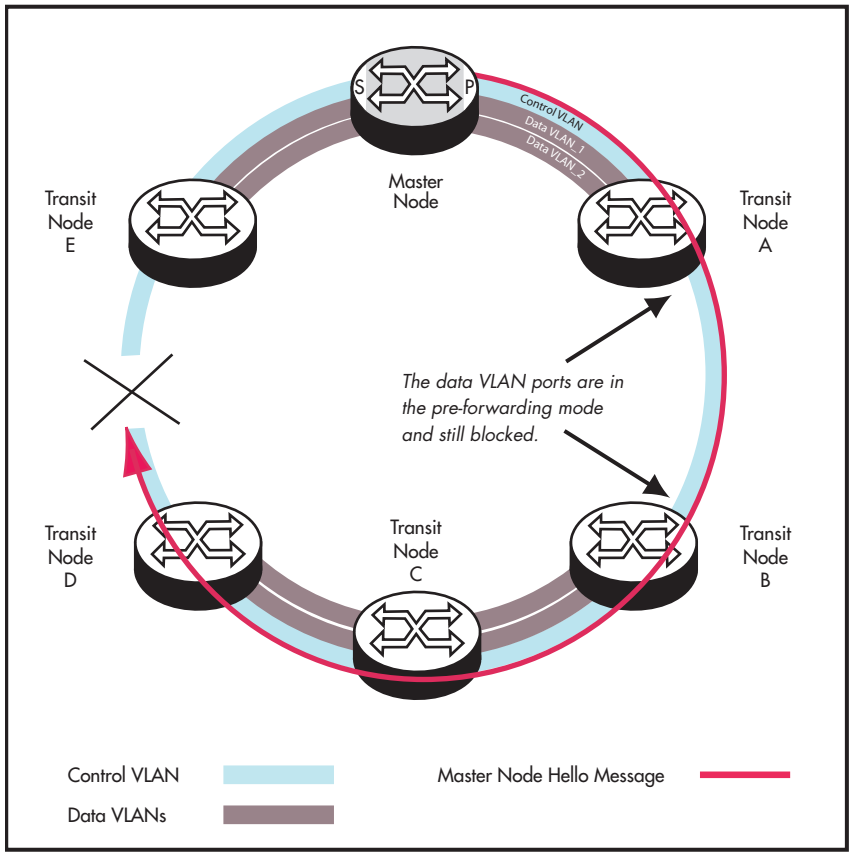


In this operational mode each portion of the ring operates as an independent link layer broadcast domain each containing the original data VLANs and control VLAN.

Recovery When One Break is Restored

Figure 83-5 on page 83.8 shows a ring with the link between nodes A and B restored. At this point the ring's behavior will depend on whether the **epsr enhancedrecovery enable** command on page 84.7 has been set.

Figure 83-5: EPSR ring with one link restored



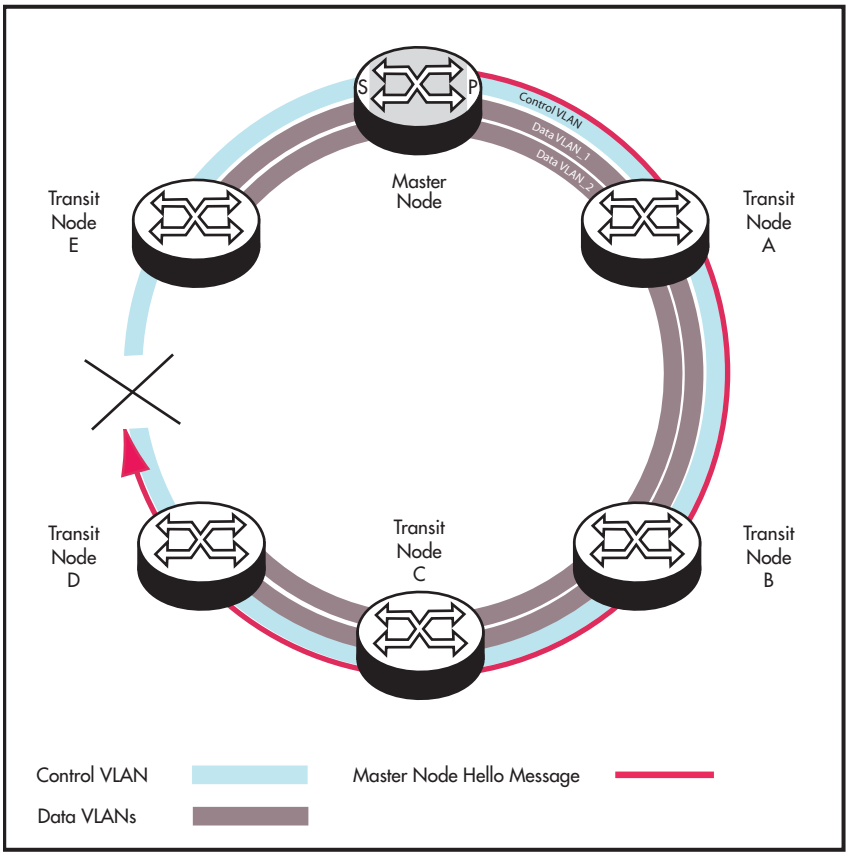
Enhanced recovery disabled

With the enhanced recovery feature disabled, the Hello messages will now reach the remaining ring break; however from a users perspective, the ring will remain as shown in the split state shown in **Figure 83-4**.

Enhanced recovery enabled

With the enhanced recovery feature enabled, switch nodes A and B are able to detect the restored link, and will place all their ring ports in the forwarding state. Although the ring will remain in the "failed" state because of the remaining break; communication between the nodes is restored. The network then operates as shown in **Figure 83-6**.

Figure 83-6: EPSR operation in partially recovered state



Configuration Examples

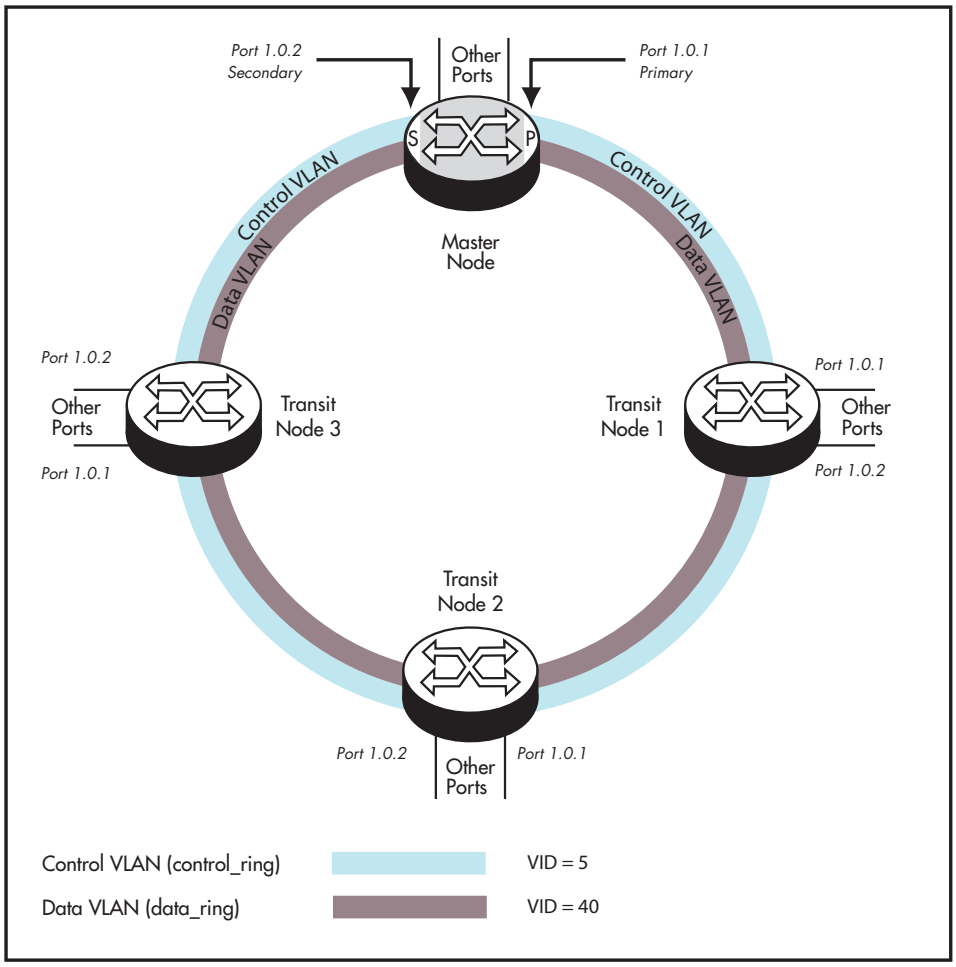
This section describes how to configure EPSR in following ways:

- **Single Domain, Single Ring Network**
- **Single Ring, Dual Domain Network**
- **EPSR and Spanning Tree Operation**

Single Domain, Single Ring Network

This example shows a simple single ring, single domain configuration with no connecting lobes.

Figure 83-7: EPSR single domain, single ring network



Configure the Master Node

Note If your switch is operating with a base user license it will function only as a transit node. To run your switch as a master node you will need to purchase a Premium License.

Step 1: Create the control and data VLANs on the Master Node

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 5 name control_vlan state enable`	Enable VLAN 5 called `control_vlan` on the Master Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `vlan 40 name data_vlan state enable`	Enable VLAN 40 called `data_vlan` on the Master Node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add port1.0.1 to these VLANs

`awplus(config)#` `interface port1.0.1`	Specify the interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#` `switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
`awplus(config-if)#` `switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#` `exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 3: Add port1.0.2 to these VLANs

`awplus(config)#`	
`interface port1.0.2`	Specify the interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 4: Create the EPSR instance called "blue" on the master node; make VLAN 5 the control VLAN and port 1.0.1 the primary port

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr blue mode master controlvlan 5 primaryport port1.0.1`	Create an EPSR instance called `blue` on `vlan5`. Make `vlan5` the control VLAN. Make `port 1.0.1` the primary port. Make this node the master. Note that you can only run this command on switches that have EPSR Master capability.

Step 5: Add a data VLAN to the EPSR instance called "blue" on the Master Node

`awplus(config-epsr)#`	
`epsr blue datavlan 40`	On epsr instance called `blue` make `vlan40` the data VLAN. Note that you can only run this command on switches that have EPSR Master capability.

Step 6: Enable the EPSR instance called "blue" on the Master Node

`awplus(config-epsr)#`	
`epsr blue state enable`	Enable the EPSR instance named `blue`. Note that you can only run this command on switches that have EPSR Master capability.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Now you can configure the transit nodes.

Step 7: Create the control and data VLANs on a transit node

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 5 name control_vlan state enable`	Enable VLAN 5 called `control_vlan` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `vlan 40 name data_vlan state enable`	Enable VLAN 40 called `data_vlan` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 8: Add port1.0.1 to the VLANs

`awplus(config)#` `interface port1.0.1`	Specify the interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#` `switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
`awplus(config-if)#` `switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#` `exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 9: Add port1.0.2 to the VLANs

`awplus(config)#`	
`interface port1.0.2`	Specify the interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 10: Create the EPSR instance called “blue” on a transit node, make VLAN 5 the control VLAN

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr blue mode transit controlvlan 5`	Create an EPSR instance called `blue` on `vlan5`. Make `vlan5` the control VLAN. Make this node a transit node.

Step 11: Add a data VLAN to the EPSR instance called “blue” on the transit node

`awplus(config-epsr)#`	
`epsr blue datavlan 40`	On the EPSR instance called `blue` make `vlan40` the data VLAN.

Step 12: Enable the EPSR instance called “blue” on the transit node

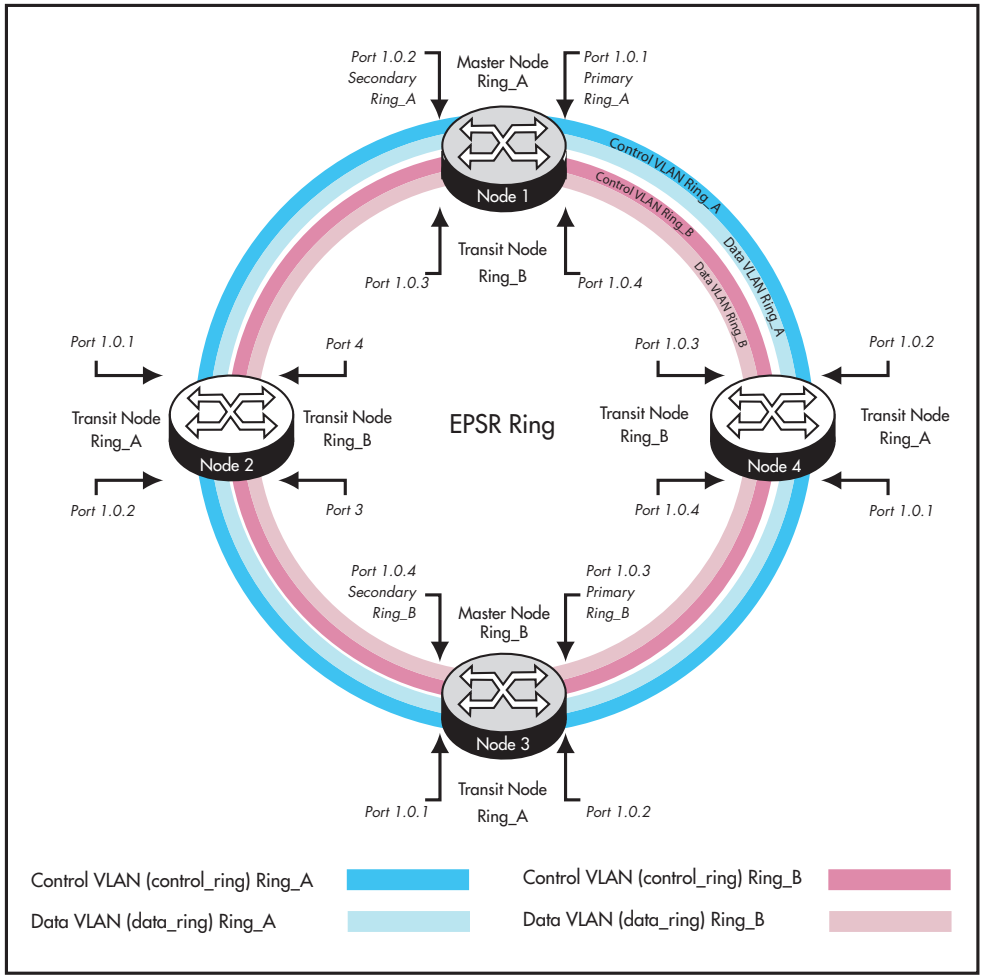
`awplus(config-epsr)#`	
`epsr blue state enable`	Enable the EPSR instance named `blue`.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Now you can use the same procedure to configure the remaining transit nodes.

Single Ring, Dual Domain Network

This example shows an EPSR configuration where two EPSR domains share the same physical ring. This configuration enables two sets of users to run totally separate Layer 2 networks. Better load distribution around the ring can be achieved by configuring different nodes to be the master for each ring.

Figure 83-8: EPSR single ring network, two domain network

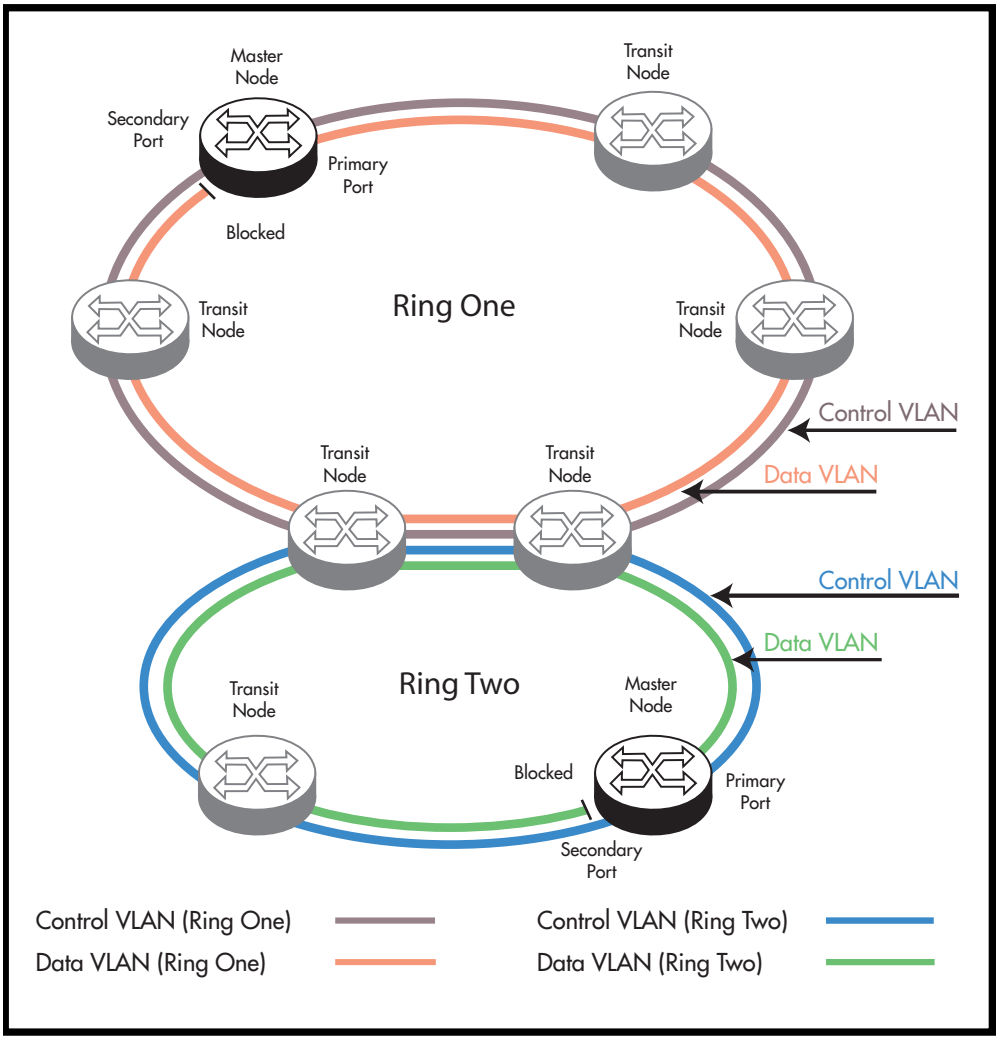


Interconnected Rings

This example shows an EPSR configuration where two rings share a common segment. This configuration will operate as two independent rings, providing that there is no data VLAN sharing between the two rings. If a break occurs in either ring then, each ring will implement its own independent recovery procedures. If a break occurs in the common segment, then each Master node will unblock its secondary port using the normal fault recovery procedure.

Where data VLANS are shared between the rings a fault condition know as “SuperLoop” can occur. The next section deals with superloops and how to manage them.

Figure 83-9: Interconnected EPSR Rings with no data VLAN sharing



Superloop Protection

Careful attention must be paid when creating EPSR networks with interconnecting links, to avoid an error condition known as superloops. This sections explains what superloops are and how to prevent them.

What is a an EPSR superloop?

An EPSR superloop is a data loop whose path traverses more than a single EPSR ring. This fault condition usually occurs when there is a break in a physical segment that is shared by the two rings. For a superloop condition to occur, the two physical rings must share some of their data VLANS. **Figure 83-10 on page 83.18**shows an EPSR ring with a superloop condition caused by a break in the common ring segment. **Figure 83-11 on page 83.19** shows the Superloop data path ring caused by the broken common ring segment. The superloop condition occurs because both rings detect the ring segment break and as a result both master nodes unblock their secondary ports.

Figure 83-10: Interconnected EPSR rings with data VLAN sharing

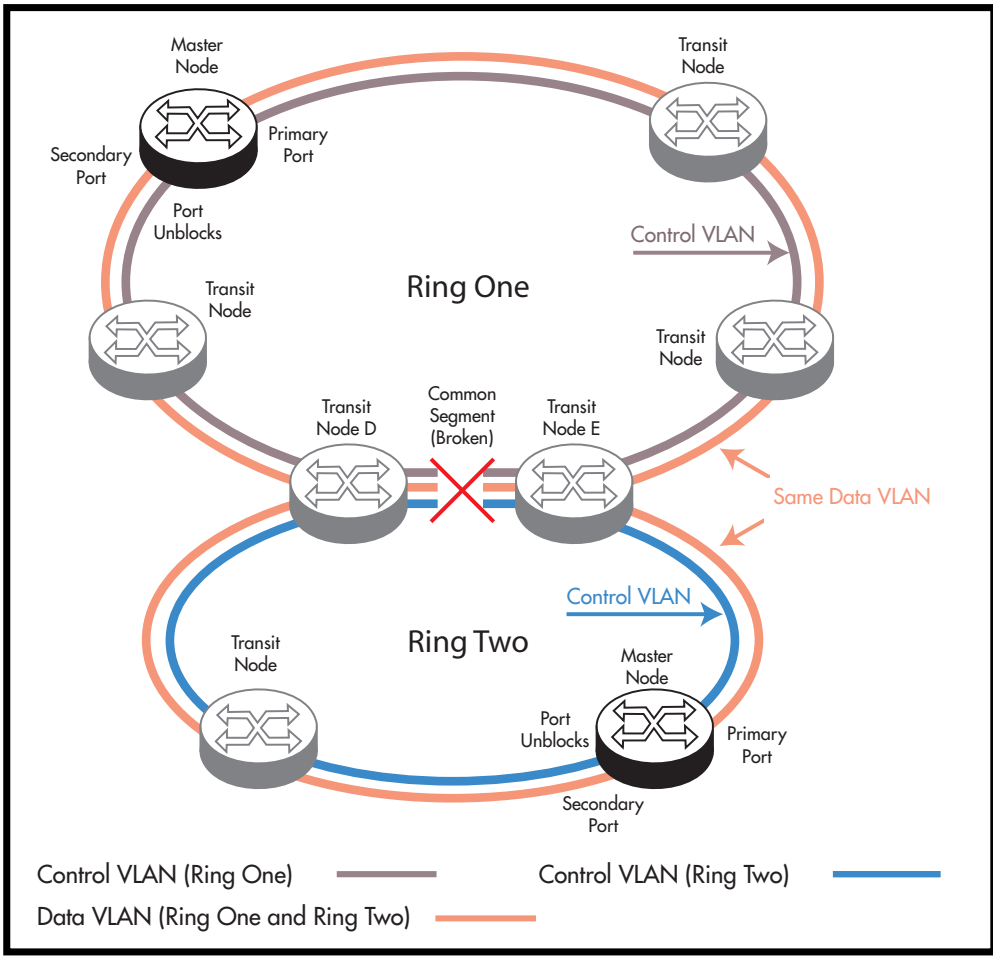
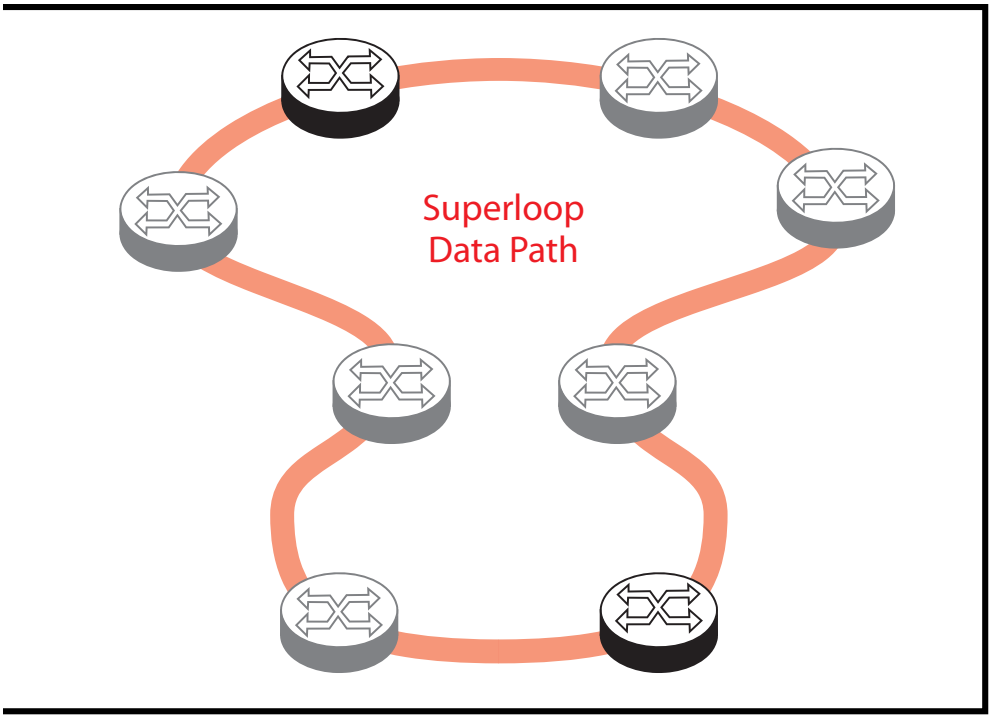


Figure 83-11: EPSR superloop data path caused by a broken common ring segment



EPSR Superloop Prevention

Alliedware Plus version 5.4.2 onwards contains mechanisms to prevent superloops forming. The Superloop prevention facility enables rings to be assigned priority level between 0 and 127, with 1 representing the lowest priority and 127 the highest. Level 0 (the default setting) applies the functionality of no Superloop prevention. Enabling superloop prevention changes the way the EPSR nodes respond under fault conditions.

Superloop prevention is enabled for an EPSR ring instance by setting the **epsr priority** command on page 84.10. Setting a priority value greater than 0 applies superloop prevention to that particular instance. How the superloop function is applied will depend on the role of the node within the ring, i.e. whether it is a master node or a transit node, and its physical location within the ring. Here is how the functions of Superloop prevention modify the nodal behavior for a particular ring instance:

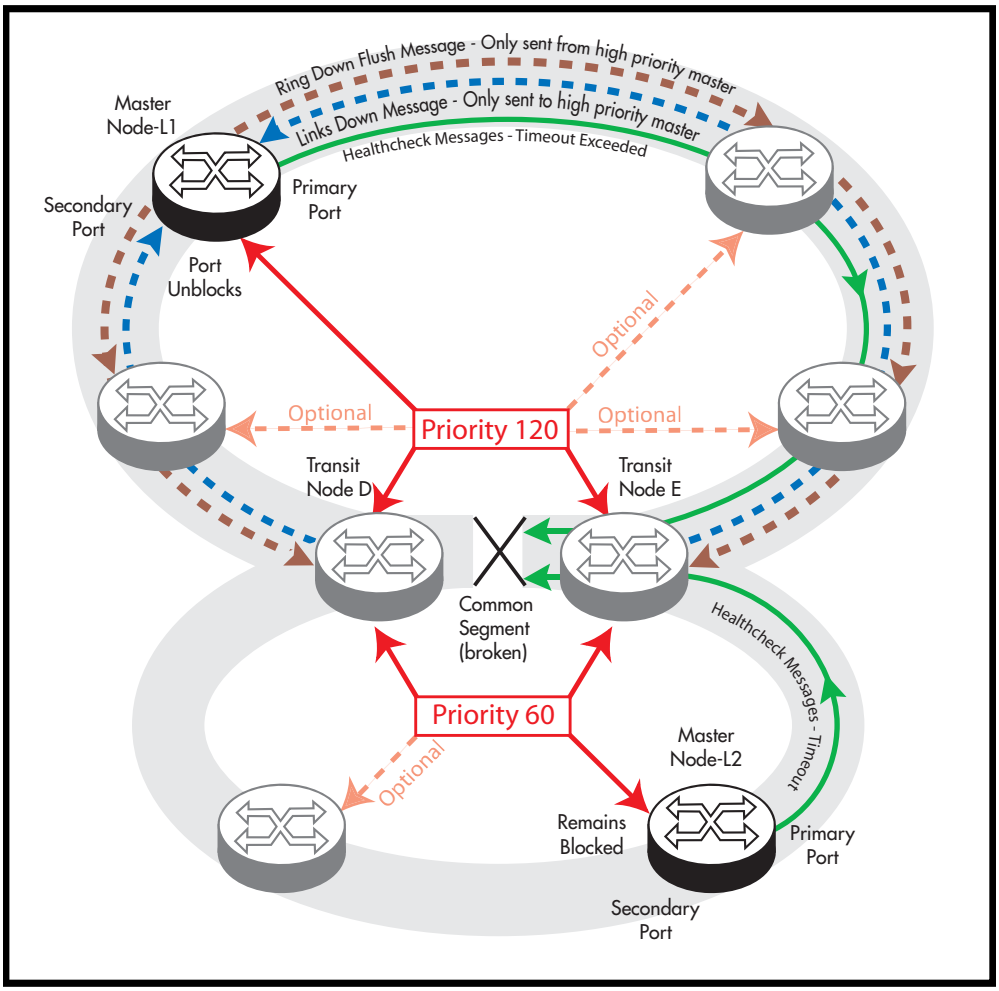
- A master node with its epsr priority set to zero will consider the superloop function to be turned off.
- A master node with its epsr priority set within the range 1-127 will consider the superloop function to be enabled, and will change its behavior in the following ways.
  - « It will **not** unblock its secondary port following the expiry of the Master Node Hello message timer. However, a ring-down-flush message will still be sent.
  - « It **will** only unblock its secondary port when it receives a Links Down message from a transit node.
- A transit node that is not connect to a shared link will be unaffected by having its epsr priority set for any of its instances.
- A transit node that is connected to a shared link will change its behavior in the following ways:
  - « It will compare its priority settings applied to each of the instances sharing the common link. So for the network of **Figure 83-10 on page 83.18** Transit Node D

will compare the priority setting for Ring One, with the priority setting for Ring Two.

If the shared link fails, the transit node will only issue a **Transit Node Links Down message** on the ring that is configured with the highest priority.

The result of these behavior changes is that when the shared link fails, only the master node located on the higher priority ring will unblock its secondary port; because this is the only master node that will receive the **Transit Node Links Down message**. Note also that the master node will receive these messages from the transit nodes at either end of the broken shared link (Nodes D and E). This concept is illustrated in the following figure.

Figure 83-12: EPSR behavior under fault conditions with superloop enabled



For this process to work certain configuration rules must be obeyed.

Configuration rules for superloop protected EPSR rings

The following configuration rules are advised when configuring EPSR rings that share one or more common segments.

- Allocate a priority order to each of the interconnected rings, with 127 being the highest priority and 1 the lowest.
- A higher priority ring can have its master node located in any position; although, where possible, avoid connecting a common segment to the secondary port of a master node.
- Do not locate the master node on a segment that is shared with a higher priority ring, but you "can" locate it on a common segment that is shared with a lower priority ring. In this situation however, the port that connects to the common segment must be configured as the primary port.

  For example, in **Figure 83-12**, the upper portion of Node D could be configured as a Master Node of the upper ring (having a priority of 120), but its lower portion must be configured as a transit node (having the lower priority of 60).
- On the transit nodes that connect to shared links, allocate the ring's priority to the ports that connect to each ring. Note that both of these nodes "must" be set to the same priority value.

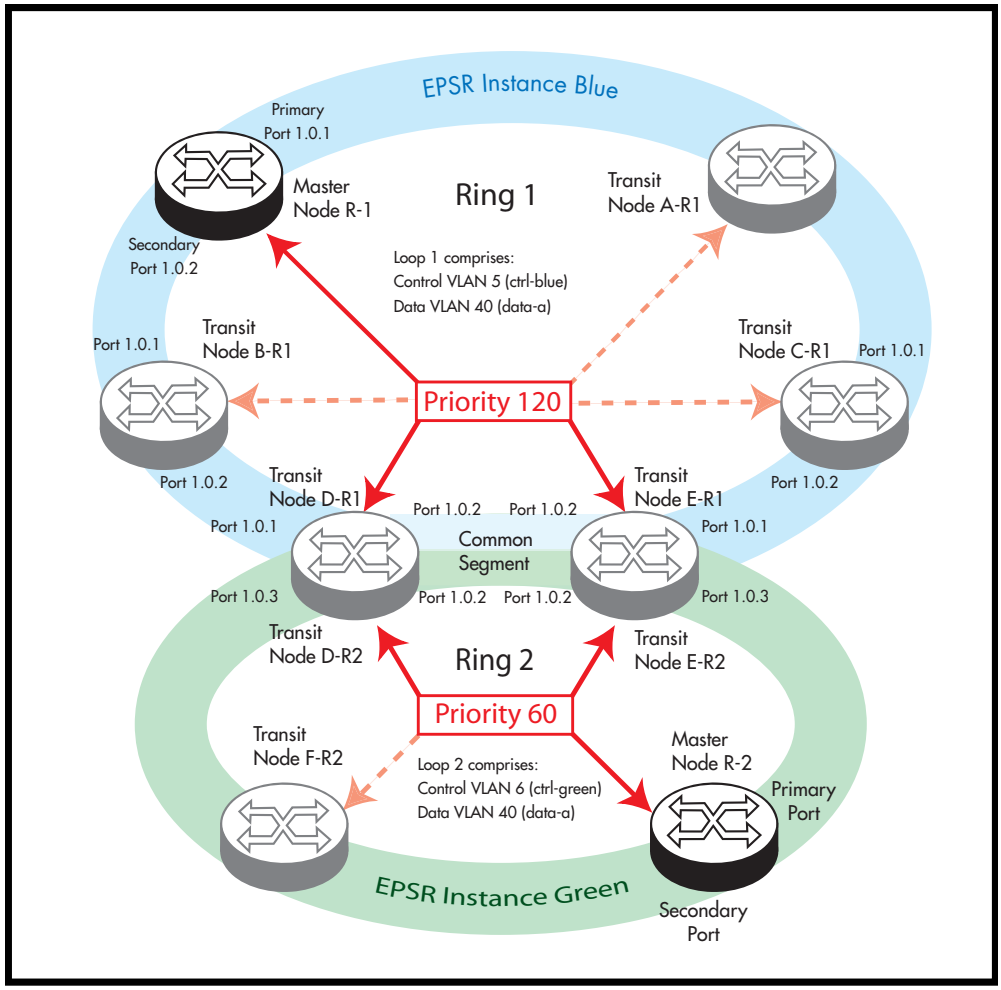
Note For good practice, we advise that you set all nodes within a ring to the priority assigned to that ring. So, for the network of **Figure 83-12** each of the nodes that form part of the upper ring would be configured with a priority of 120, and each of the nodes that form the lower ring would all be configured with a priority of 60.

Configuring a Basic Superloop Protected Two Ring EPSR Network

Configuration example

This section shows how to configure a basic EPSR network such as that shown in the figure below.

Figure 83-13: EPSR two shared ring example



The configuration suggested comprises the following basic steps:

- **"On Ring 1- configure the Master Node R-1" on page 83.23**
- **"On Ring 1 - configure the transit nodes A to C" on page 83.25**
- **"On Ring 2 - configure the Master Node R-2" on page 83.27**
- **"On Rings 1 and 2 - configure the Transit Nodes D and E" on page 83.29**
- **"On Ring 2 - configure the Transit Node F" on page 83.35**

Note If your switch is operating with a base user license it will function only as a transit node. To run your switch as a master node you will need to purchase a Premium License.

On Ring 1- configure the Master Node R-1

Step 1: Create the control and data VLANs (configure on the Master Node R-1)

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 5 name ctrl-blue state enable`	Enable VLAN 5 called ctrl-blue on the Master Node R-1. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `vlan 40 name data-a state enable`	Enable VLAN 40 called **data-a** on the Master Node R-1. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add the control VLAN (ctrl-blue) to the ring ports

`awplus(config)#` `interface port1.0.1,port1.0.2`	Specify the two ring ports (`port1.0.1 and port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of these ports to Trunk mode.
`awplus(config-if)#` `switchport trunk allowed vlan add 5`	Enable VLAN 5 on these ports.
`awplus(config-if)#` `switchport trunk native vlan none`	Remove the native VLAN from these ring ports.

Step 3: Create the EPSR instance called "blue", make VLAN 5 the control VLAN and port 1.0.1 the primary port (configure on the Master Node R-1)

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr blue mode master controlvlan 5 primaryport port1.0.1`	Create an EPSR instance called `blue` on `vlan 5`. Make `vlan 5` the control VLAN. Make `port 1.0.1` the primary port. Make this node the master.

Step 4: Add a data VLAN to the EPSR instance called "blue" (Configure on the Master Node R-1)

`awplus(config-epsr)#`	
`epsr blue datavlan 40`	On epsr instance called blue data-a the data VLAN.

Step 5: Assign a priority to the ring instance (Configure on the Master Node R-1)

`awplus(config-epsr)#`	
`epsr blue priority 120`	Set the ring instance priority to the value selected for the ring. The priority value selected is 120.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Step 6: Enable the EPSR instance called "blue" (configure on the Master Node R-1)

`awplus(config-epsr)#`	
`epsr blue state enable`	Enable the EPSR instance named `blue`.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Step 7: Add port1.0.1 to these VLANs (configure on the Master Node R-1)

`awplus(config)#`	
`interface port1.0.1,port1.0.2`	Specify the EPSR ring ports (`port1.0.1 and 1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

On Ring 1 - configure the transit nodes A to C

Step 1: Create the control and data VLANs (on Transit Nodes A to C)

`awplus#` `configure terminal`	Enter the Global Configuration mode.
`awplus(config)#` `vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#` `vlan 5 name ctrl-blue state enable`	Enable VLAN 5 called `ctrl-blue` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `vlan 40 name data-a state enable`	Enable VLAN 40 called `data-a` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#` `exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add the EPSR control vlan (ctrl-blue) to EPSR ring ports

`awplus(config)#` `interface port1.0.1,port1.0.2`	Specify the two ring ports (`port1.0.1 and port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#` `switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#` `switchport trunk allowed vlan add 5`	Enable VLAN 5 on these ports.
`awplus(config-if)#` `switchport trunk native vlan none`	Remove the native VLAN from the ring ports.

Step 3: Create the EPSR instance called "blue", make VLAN 5 the control VLAN (on Transit Nodes A to C)

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr blue mode transit controlvlan 5`	Create an EPSR instance called `blue` on `vlan 5`. Make `vlan 5` the control VLAN. Make this node a transit node.

Step 4: Add a data VLAN to the EPSR instance called "blue" (on Transit Nodes A to C)

`awplus(config-epsr)#`	
`epsr blue datavlan 40`	On the EPSR instance called `blue` make `vlan 40` the data VLAN.

Step 5: Assign a priority to the ring instance (on Transit Nodes A to C)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

`awplus(config-epsr)#`	
`epsr blue priority 120`	Set the ring instance priority to the priority selected for the ring 120.

Step 6: Enable the EPSR instance called "blue" (on Transit Nodes A to C)

`awplus(config-epsr)#`	
`epsr blue state enable`	Enable the EPSR instance named `blue`.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Step 7: Add the physical ports 1.0.1 and 1.0.2 to VLAN 40 (on Transit Nodes A to C)

`awplus(config)#`	
`interface port1.0.1,port1.0.2`	Specify the physical ring ports (`ports1.0.1` and ports 1.0.2) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on these ports.
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

On Ring 2 - configure the Master Node R-2

Step 1: Create the control and data VLANs (configure on the Master Node R-2)

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#`	
`vlan 6 name ctrl-green state enable`	Enable vlan 6 called ctrl-green on the Master Node R-2. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`vlan 40 name data-a state enable`	Enable VLAN 40 called data-a on the Master Node R-2. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add the control VLAN (ctrl-green) to the ring ports

`awplus(config)#`	
`interface port1.0.1,port1.0.2`	Specify the ports (`port1.0.1 and port1.0.2`) that you are configuring, and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of these ports to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 6`	Enable vlan 6 on these ports.
`awplus(config-if)#`	
`switchport trunk native vlan none`	Remove the native VLAN from these ring ports.

Step 3: Create the EPSR instance called "green", make VLAN 6 the control VLAN and port1.0.1 the primary port (configure on the Master Node R-2)

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr green mode master controlvlan 6 primaryport port1.0.1`	Create an EPSR instance called `ctrl-green` on `vlan 6`. Make `vlan 6` the control VLAN. Make `port 1.0.1` the primary port. Make this node the master.

Step 4: Add a data VLAN to the EPSR instance "green" (configure on the Master Node R-2)

`awplus(config-epsr)#`	
`epsr green datavlan 40`	On epsr instance called green make `vlan 40` the data VLAN.

Step 5: Assign a priority to the ring instance (configure on the Master Node R-2)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

`awplus(config-epsr)#`	
`epsr green priority 60`	Set the ring instance priority to the value selected for the ring. The priority value selected is 60.

Step 6: Enable the EPSR instance called "green" (configure on the Master Node R-2)

`awplus(config-epsr)#`	
`epsr green state enable`	Enable the EPSR instance named green.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Step 7: Add ports 1.0.1 and 1.0.2 to these VLANs (configure on the Master Node R-2)

`awplus(config)#`	
`interface port1.0.1,port1.0.2`	Specify the ports (`port1.0.1 and port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of these ports to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

On Rings 1 and 2 - configure the Transit Nodes D and E

Step 1: Create the control and data VLANs (on Transit Nodes D and E)

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#`	
`vlan 5 name ctrl-blue state enable`	Enable VLAN 5 called `ctrl-blue` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`vlan 40 name data-a state enable`	Enable VLAN 40 called `data-a` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`vlan 6 name ctrl-green state enable`	Enable VLAN 6 called `ctrl-green` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Add physical port1.0.1 to these VLANs (on Transit Nodes D and E)

`awplus(config)#`	
`interface port1.0.1`	Specify the physical `interface (port1.0.1)` that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 5`	Enable VLAN 5 on this port.
`awplus(config-if)#`	
`switchport trunk native vlan none`	Remove the native VLAN.
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 3: Add physical port1.0.2 to these VLANs (on Transit Nodes D and E)

`awplus(config)#`	
`interface port1.0.2`	Specify the physical `interface (port1.0.2)` that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 5`	Enable VLAN 5 (ctrl-blue) on this port.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 6`	Enable VLAN 6 (ctrl-green) on this port.
`awplus(config-if)#`	
`switchport trunk native vlan none`	Remove the native VLAN.
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 4: Add physical port1.0.3 to these VLANs (on Transit Nodes D and E)

`awplus(config)#`	
`interface port1.0.3`	Specify the physical `interface (port1.0.3)` that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.

`awplus(config-if)#`	
`switchport trunk allowed vlan add 6`	Enable VLAN 6 on this port.
`awplus(config-if)#`	
`switchport trunk native vlan none`	Remove the native VLAN.
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 5: Create the EPSR instance called "blue" on a transit node, make VLAN 5 the control VLAN (on Transit Nodes D and E)

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr blue mode transit controlvlan 5`	Create an EPSR instance called `blue` on `vlan 5`. Make `vlan 5` the control VLAN. Make this node a transit node.

Step 6: Add a data VLAN to the EPSR instance called "blue" (on Transit Nodes D and E)

`awplus(config-epsr)#`	
`epsr blue datavlan 40`	On the EPSR instance called `blue` make `vlan 40` the data VLAN.

Step 7: Assign a priority to the ring instance (on Transit Nodes D and E)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

`awplus(config-epsr)#`	
`epsr blue priority 120`	Set the ring instance priority to 120 - the value selected for the ring.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Step 8: Enable the EPSR instance called "blue" (on Transit Nodes D and E)

`awplus(config-epsr)#`	
`epsr blue state enable`	Enable the EPSR instance named `blue`.

Step 9: Create the EPSR instance called "green" on a transit node, make VLAN 6 the control VLAN (on Transit Nodes D and E)

`awplus(config-epsr)#`	
`epsr green mode transit controlvlan 6`	Create an EPSR instance called `green` on `vlan 6`. Make `vlan 6` the control VLAN. Make this node a transit node.

Step 10: Add a data VLAN to the EPSR instance called "green" (on Transit Nodes D and E)

`awplus(config-epsr)#`	
`epsr green datavlan 40`	On the EPSR instance called `green` make `vlan 40` the data VLAN.

Step 11: Assign a priority to the ring instances (on Transit Nodes D and E)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

```
awplus(config-epsr)#
epsr green priority 60
```
Set the ring instance priority to 60 - this being the priority selected for the ring.

```
awplus(config-epsr)#
exit
```
Exit the EPSR Configuration mode.

Step 12: Enable the EPSR instance called "green" (on Transit Nodes D and E)

```
awplus(config-epsr)#
epsr green state enable
```
Enable the EPSR instance named `green`.

```
awplus(config-epsr)#
exit
```
Exit the EPSR Configuration mode.

Step 13: Add the physical port 1.0.1 to these VLANs (on Transit Nodes D and E)

```
awplus(config)#
interface port1.0.1
```
Specify the physical interface (`port1.0.1`) that you are configuring, and enter the Interface Configuration mode.

```
awplus(config-if)#
switchport mode trunk
```
Set the switching characteristics of this port to Trunk mode.

```
awplus(config-if)#
switchport trunk allowed vlan add 40
```
Enable VLAN 40 on this port.

```
awplus(config-if)#
exit
```
Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 14: Add the physical port1.0.2 to these VLANs (on Transit Nodes D and E)

`awplus(config)#`	
`interface port1.0.2`	Specify the physical interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 15: Add the physical port1.0.3 to these VLANs (on Transit Nodes D and E)

`awplus(config)#`	
`interface port1.0.3`	Specify the physical interface (`port1.0.3`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

On Ring 2 - configure the Transit Node F

Step 1: Create the control and data VLANs (on Transit Node F)

`awplus#`	
`configure terminal`	Enter the Global Configuration mode.
`awplus(config)#`	
`vlan database`	Enter the VLAN Configuration mode.
`awplus(config-vlan)#`	
`vlan 6 name ctrl-green state enable`	Enable VLAN 6 called `ctrl-green` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`vlan 40 name data-a state enable`	Enable VLAN 40 called `data-a` on the transit node. Specifying the enable state allows forwarding of frames on the VLAN-aware node.
`awplus(config-vlan)#`	
`exit`	Exit the VLAN Configuration mode and enter the Global Configuration mode.

Step 2: Create the EPSR instance called "green" on a transit node, make VLAN 6 the control VLAN (on Transit Node F)

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr green mode transit controlvlan 6`	Create an EPSR instance called `green` on `vlan 6`. Make `vlan 6` the control VLAN. Make this node a transit node.

Step 3: Add a data VLAN to the EPSR instance called "green" (on Transit Node F)

`awplus(config-epsr)#`	
`epsr green datavlan 40`	On the EPSR instance called `green` make `vlan 40` the data VLAN.

Step 4: Enable the EPSR instance called "green" (on Transit Node F)

`awplus(config-epsr)#`	
`epsr green state enable`	Enable the EPSR instance named `green`.

Step 5: Assign a priority to the ring instance (on Transit Node F)

This step is **mandatory on transit nodes that connect to a common segment**, and good practice on other transit nodes.

`awplus(config-epsr)#`	
`epsr green priority 120`	Set the ring instance priority to the priority selected for the ring 120.
`awplus(config-epsr)#`	
`exit`	Exit the EPSR Configuration mode.

Step 6: Add the physical port1.0.1 to VLANs 6 and 40 (on Transit Node F)

`awplus(config)#`	
`interface port1.0.1`	Specify the physical interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 6`	Enable VLAN 6 on this port.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`switchport trunk native vlan none`	Remove the native VLAN
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 7: Add the physical port1.0.2 to VLANs 6 and 40 (on Transit Node F)

`awplus(config)#`	
`interface port1.0.2`	Specify the interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport mode trunk`	Set the switching characteristics of this port to Trunk mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 6`	Enable VLAN 6 on this port.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 40`	Enable VLAN 40 on this port.
`awplus(config-if)#`	
`switchport trunk native vlan none`	Remove the native VLAN
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Sample Show Output

For the above network configuration, running the command **show epsr** on node R1 will display the following output when operating normally. Note the blocked state of its secondary port.

Figure 83-14: Output from the show epsr command run on Master Node R1 - with Ring 1 - EPSR instance blue operating normally

```
 EPSR Information
-------------------------------------------------------------------
 Name .....................blue
 Mode .....................Master
 Status ...................Enabled
 State ....................Complete
 Control Vlan .............5
 Data VLAN(s) .............40
 Interface Mode ...........Ports Only
 Primary Port .............port1.0.1
   Status .................Forwarding
   Is On Common Segment ...No
   Blocking Control .......Physical
 Secondary Port ...........port1.0.2
   Status .................Blocked
   Is On Common Segment ...No
   Blocking Control .......Physical
 Hello Time ...............1 s
 Failover Time ............2 s
 Ring Flap Time ...........0 s
 Trap .....................Enabled
 Enhanced Recovery ........Disabled
 Priority .................120
-------------------------------------------------------------------
```

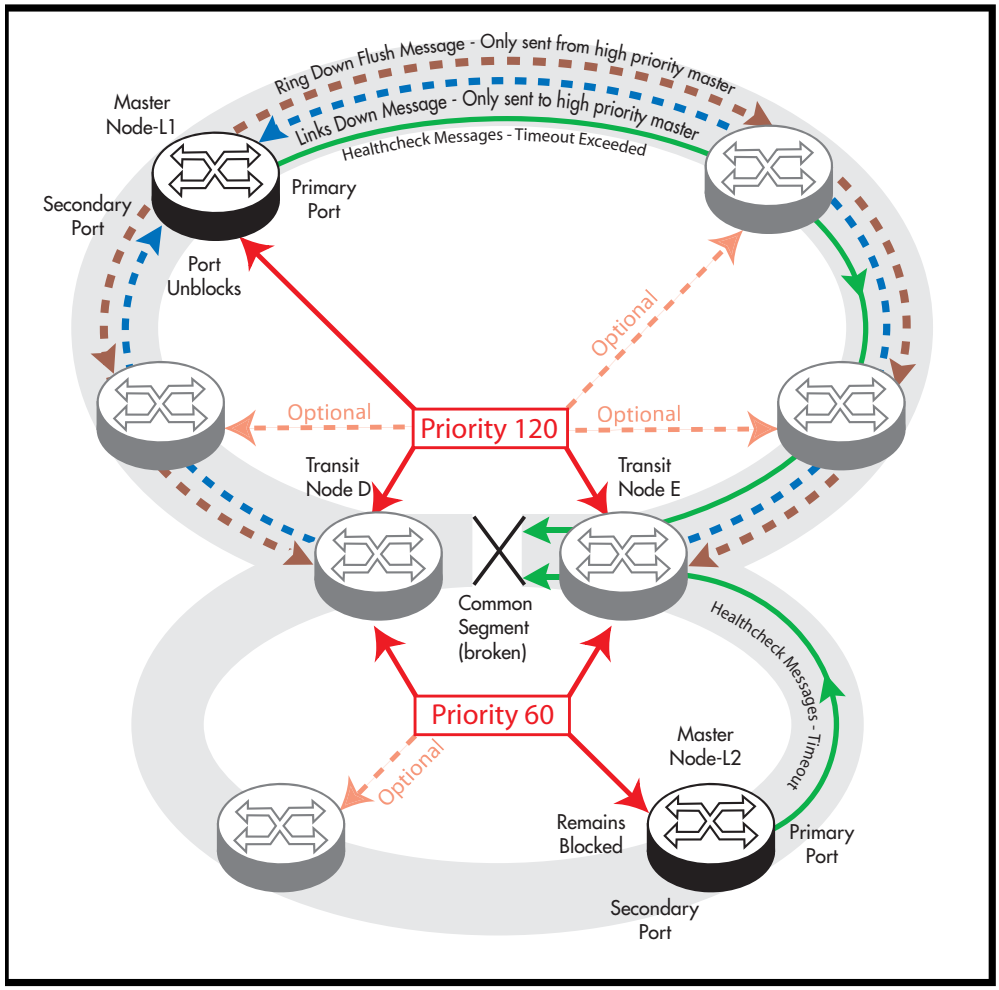
If a fault occurs somewhere within the blue network ring the Master Node-R1 would respond by placing its secondary port into the forwarding state. Figure **Figure 83-15** displays its resultant state. Note that the state of its secondary port has now moved from Blocked, Forwarding.

Figure 83-15: Output from the show epsr command run on Master Node R2, where a break exists within the Ring 1 - EPSR instance blue

```
 EPSR Information
-------------------------------------------------------------------
 Name .....................blue
 Mode .....................Master
 Status ...................Enabled
 State ....................Failed
 Control Vlan .............6
 Data VLAN(s) .............40
 Interface Mode ...........Ports Only
 Primary Port .............port1.0.1
   Status .................Forwarding
   Is On Common Segment ...No
   Blocking Control .......Physical
 Secondary Port ...........port1.0.2
   Status .................Forwarding
   Is On Common Segment ...No
   Blocking Control .......Physical
 Hello Time ...............1 s
 Failover Time ............2 s
 Ring Flap Time ...........0 s
 Trap .....................Enabled
 Enhanced Recovery ........Disabled
 Priority .................60
-------------------------------------------------------------------
```

If a fault occurs in the common segment of the ring then the Master Node-R2 being on the lower priority ring would detect a timeout of its transmitted Healthcheck Message. It would also detect the absence of the expected **Ring Down Flush** message, see **Figure 83-16**. The Master node then assumes that there is a break somewhere in the Common Segment, and will display the status shown in **Figure 83-17**.

Figure 83-16: EPSR behavior with a faulty common segment and superloop enabled



Note that the secondary port on Master Node-L2 remains in the blocked state; its state now appears in show output as being as blocked (for superloop prevention). See **Figure 83-17**.

The Master-L1 on the blue ring will also detect a timeout in the healthcheck message, but because ring 1 has the higher priority (of 120), it will receive a Links Down message from each of the transit nodes (D and E) that connect to the common segment. As a result, the state of the Master Node will be as shown in **Figure 83-17**; note particularly the change in its Secondary Port status.

Figure 83-17: Output from the show epsr command run on Master Node L2 (green)

```
 EPSR Information
-------------------------------------------------------------------
 Name .....................green
 Mode .....................Master
 Status ...................Enabled
 State ....................Failed
 Control Vlan .............6
 Data VLAN(s) .............40
 Interface Mode ...........Ports Only
 Primary Port .............port1.0.1
   Status .................Forwarding
   Is On Common Segment ...No
   Blocking Control .......Physical
 Secondary Port ...........port1.0.2
   Status .................Blocked (for superloop prevention)
   Is On Common Segment ...No
   Blocking Control .......Physical
 Hello Time ...............1 s
 Failover Time ............2 s
 Ring Flap Time ...........0 s
 Trap .....................Enabled
 Enhanced Recovery ........Disabled
 Priority .................60
-------------------------------------------------------------------
```

Figure 83-18: Output from the show epsr command run on Master Node L1 (blue)

```
 EPSR Information
-------------------------------------------------------------------
 Name .....................blue
 Mode .....................Master
 Status ...................Enabled
 State ....................Failed
 Control Vlan .............6
 Data VLAN(s) .............40
 Interface Mode ...........Ports Only
 Primary Port .............port1.0.1
   Status .................Forwarding
   Is On Common Segment ...No
   Blocking Control .......Physical
 Secondary Port ...........port1.0.2
   Status .................Forwarding
   Is On Common Segment ...No
   Blocking Control .......Physical
 Hello Time ...............1 s
 Failover Time ............2 s
 Ring Flap Time ...........0 s
 Trap .....................Enabled
 Enhanced Recovery ........Disabled
 Priority .................60
-------------------------------------------------------------------
```

Adding a New Data VLAN to a Functioning Superloop Topology

This example shows how to add another data VLAN called **data-b** to the superloop topology. We recommend that you apply the configuration steps in the order shown.

1. Add VLAN to the common segment (for both instances)
2. Add VLAN to blue master
3. Add VLAN to other blue transits
4. Add VLAN to green master
5. Add VLAN to other green transits

On Ring 1 EPSR instance blue - configure each of the transit nodes that connect to the common segment

Select one of the transit nodes that connects to the common segment, and carry out the following steps:

Step 1: Add VLAN 50 to the VLAN database and set its state to enable

`awplus#`	
`configure terminal`	Enter terminal config mode
`awplus(config)#`	
`vlan database`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`vlan 50 name data-b enable`	Create VLAN 50, name it data-b and enable it.

Step 2: Add the VLAN 50 to the EPSR Instances called “blue” and “green” on the transit nodes

`awplus(config)#`	
`epsr configuration`	Enter the EPSR Configuration mode.
`awplus(config-epsr)#`	
`epsr blue datavlan 50`	On the EPSR instance called `blue` add `vlan 50` as a data VLAN.
`awplus(config-epsr)#`	
`epsr green datavlan 50`	On the EPSR instance called `green` add `vlan 50` as a data VLAN.

Step 3: Add the common physical port (port1.0.2 in this example) to VLAN 50

`awplus(config)#`	
`interface port1.0.2`	Specify the physical interface (`port1.0.2`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 50`	Enable VLAN 50 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface mode and enter the Global Configuration mode.

Step 4: Add physical port1.0.1 to VLAN 50

`awplus(config)#`	
`interface port1.0.1`	Specify the interface (`port1.0.1`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 50`	Enable VLAN 50 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Step 5: Add physical port1.0.3 to VLAN 50

`awplus(config)#`	
`interface port1.0.3`	Specify the interface (`port1.0.3`) that you are configuring and enter the Interface Configuration mode.
`awplus(config-if)#`	
`switchport trunk allowed vlan add 50`	Enable VLAN 50 on this port.
`awplus(config-if)#`	
`exit`	Exit the Interface Configuration mode and enter the Global Configuration mode.

Select the next transit node that connects to the common segment, and repeat the above steps:

On Ring 1 EPSR instance blue - Add VLAN 50 to the Master Node

Carry out this process using the same basic procedure shown in Steps 1 to 5

On Ring 1 EPSR instance blue - Add VLAN 50 to the transit nodes

Carry out this process using the same basic procedure shown in Steps 1 to 5

On Ring 2 EPSR instance green - Add VLAN 50 to the Master Node

Carry out this process using the same basic procedure shown in Steps 1 to 5

On Ring 2 EPSR instance green - Add VLAN 50 to the remaining transit node

Carry out this process using the same basic procedure shown in Steps 1 to 5

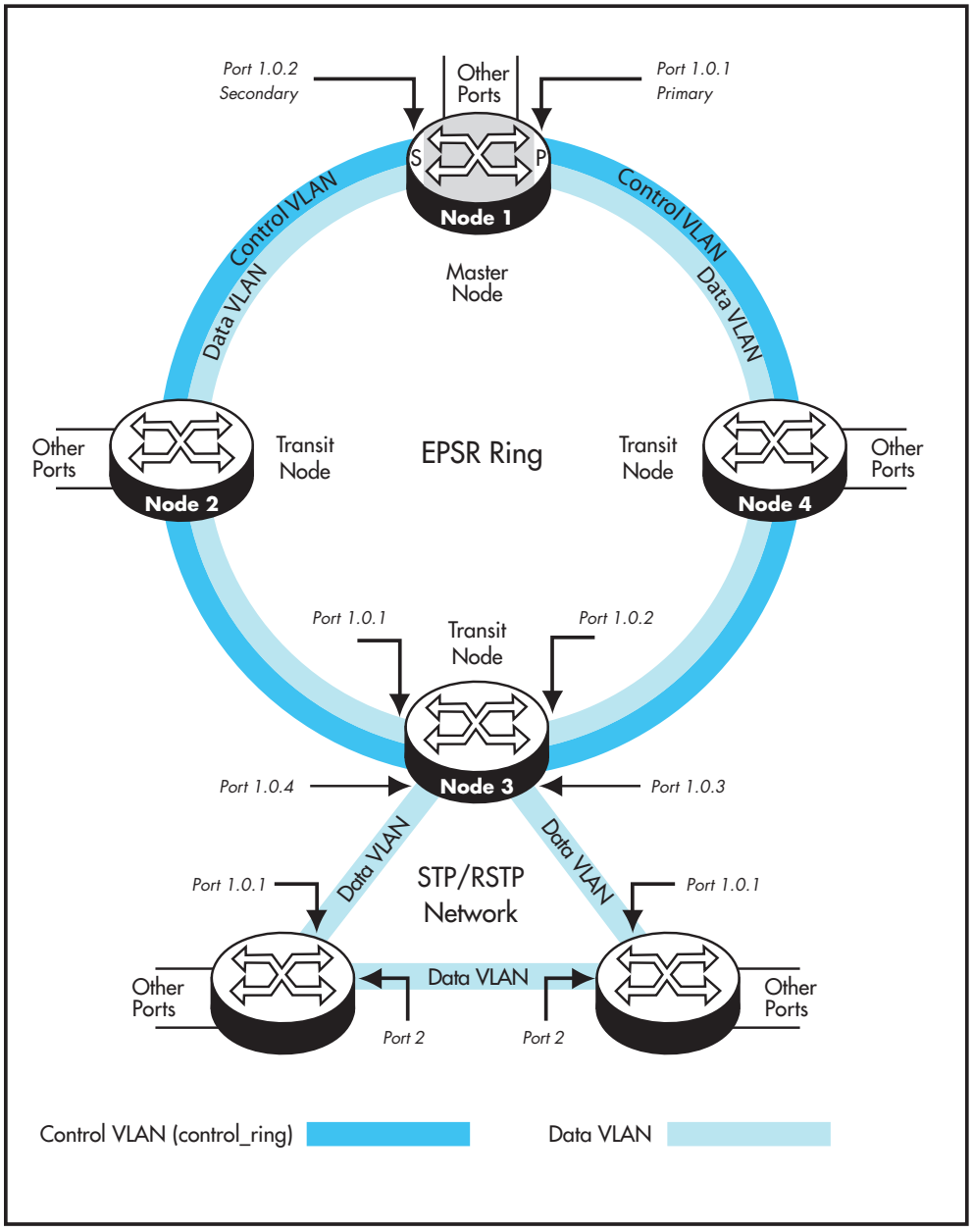
EPSR and Spanning Tree Operation

EPSR and the Spanning Tree Protocol (STP) address data loop prevention, although they do it differently. EPSR is manually configured to explicitly identify which links are broken in the defined ring, whereas STP/RSTP calculates where to break links based on user-provided values (metrics) that are compared to determine the “best” (or lowest cost) paths for data traffic.

At the practical level you can use these two techniques to create complementary hybrid EPSR /STP configurations. This configuration might have a high speed fiber loop topology backbone-controlled and managed using EPSR. Lobes could extend out from each loop node into a user mesh network. Any loops in this mesh network would be controlled and managed using STP/RSTP. Note that EPSR and STP cannot share the same ports.

The following figure shows a basic combined EPSR / STP network.

Figure 83-19: EPSR and spanning tree operation



Chapter 84: EPSR Commands

Command List 84.2
debug epsr 84.3
epsr 84.4
epsr configuration 84.5
epsr datavlan 84.6
epsr enhancedrecovery enable 84.7
epsr mode master controlvlan primary port 84.8
epsr mode transit controlvlan 84.9
epsr priority 84.10
epsr state 84.11
epsr trap 84.12
show debugging epsr 84.12
show epsr 84.13
show epsr common segments 84.18
show epsr config-check 84.19
show epsr <epsr-instance> 84.20
show <epsr-instance> counters 84.20
show epsr counters 84.21
show epsr summary 84.22
undebug epsr 84.22

Command List

This chapter provides an alphabetical reference for commands used to configure EPSR. For more information, see **Chapter 83, EPSR Introduction and Configuration**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

debug epsr

This command enables EPSR debugging.

The **no** variant of this command disables EPSR debugging.

Syntax

```
debug epsr {info|msg|pkt|state|timer|all}
no debug epsr {info|msg|pkt|state|timer|all}
```

Parameter	Description
`info`	Send general EPSR information to the console. Using this parameter with the **no debug epsr** command will explicitly exclude the above information from being sent to the console.
`msg`	Send the decoded received and transmitted EPSR packets to the console. Using this parameter with the **no debug epsr** command will explicitly exclude the above packets from being sent to the console.
`pkt`	Send the received and transmitted EPSR packets as raw ASCII text to the console. Using this parameter with the **no debug epsr** command will explicitly exclude the above packets from being sent to the console.
`state`	Send EPSR state transitions to the console. Using this parameter with the **no debug epsr** command will explicitly exclude state transitions from being sent to the console.
`timer`	Send EPSR timer information to the console. Using this parameter with the **no debug epsr** command will explicitly exclude timer information from being sent to the console.
`all`	Send all EPSR debugging information to the console. Using this parameter with the **no debug epsr** command will explicitly exclude any debugging information from being sent to the console.

Mode Privileged Exec and Global Configuration

Examples To enable state transition debugging, use the command:

```
awplus# debug epsr state
```

To disable EPSR packet debugging, use the command:

```
awplus# no debug epsr pkt
```

Related Commands undebug epsr

epsr

This command sets the timer values for an EPSR instance. These are only valid for master nodes.

Note Running your switch as an EPSR master node requires a Premium License.

Syntax

```
epsr <epsr-instance> {hellotime <1-32767>|failovertime <2-65535>
   ringflaptime <0-65535>}

no epsr <epsr-instance>
```

Caution **Using the "no" variant of this command will remove the specified EPSR instance.**

Parameter	Description
<epsr-instance>	Name of the EPSR instance.
`hellotime` *<1-32767>*	The number of seconds between the transmission of health check messages.
`failovertime` *<2-65535>*	The number of seconds that a master waits for a returning health check message before entering the failed state. **The failover time should be greater than twice the hellotime.** This is to force the master node to wait until it detects the absence of two sequential healthcheck messages before entering the failed state.
`ringflaptime` *<0-65535>*	The minimum number of seconds that a master must remain in the failed state.

Mode EPSR Configuration

Examples To set the hellotimer to 5 seconds for the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue hellotime 5
```

Note When stacking is used with EPSR, the EPSR **failovertime** should be at least 5 seconds.

To delete the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue
```

Related Commands

epsr mode master controlvlan primary port
epsr mode transit controlvlan
epsr configuration
epsr datavlan
epsr state
epsr trap
reboot rolling
show epsr

epsr configuration

Use this command to enter EPSR Configuration mode so that EPSR can be configured.

Syntax

```
epsr configuration
```

Mode

Global Configuration

Example

To change to EPSR mode, use the command:

```
awplus(config)# epsr configuration
```

Related Commands

epsr mode master controlvlan primary port
epsr
show epsr

epsr datavlan

This command adds a data VLAN or a range of VLAN identifiers to a specified EPSR instance.

The **no** variant of this command removes a data vlan or data vlan range from an EPSR instance.

Syntax

```
epsr <epsr-instance> datavlan {<vlanid>|<vlanid-range>}
no epsr <epsr-instance> datavlan {<vlanid>|<vlanid-range>}
```

Parameter	Description
<epsr-instance>	Name of the EPSR instance.
datavlan	Adds a data VLAN to be protected by the EPSR instance.
<vlanid>	The VLAN's VID - a number between 1 and 4094 excluding the number selected for the control VLAN.
<vlanid-range>	Specify a range of VLAN identifiers using a hyphen to separate identifiers.

Mode EPSR Configuration

Usage We suggest setting the epsr controlvlan to vlan2 using the **epsr mode master controlvlan primary port** and **epsr mode transit controlvlan** commands, then setting the EPSR data VLAN between to be a value 3 and 4094 using the **epsr datavlan** command.

Examples To add `vlan3` to the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue datavlan vlan3
```

To add `vlan2` and `vlan3` to the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue datavlan vlan2-vlan3
```

To remove `vlan3` from the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue datavlan vlan3
```

To remove `vlan2` and `vlan3` from the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue datavlan vlan2-vlan3
```

Related Commands

epsr mode master controlvlan primary port
epsr mode transit controlvlan
show epsr

epsr enhancedrecovery enable

This command enables EPSR's enhanced recovery mode. Enhanced recovery mode enables a ring to apply additional recovery procedures when a ring with more than one break, partially mends. For more information see **"Managing Rings with Two Breaks" on page 83.6**.

The **no** variant of this command disables the enhancedrecovery mode.

Syntax

```
epsr <epsr-instance> enhancedrecovery enable
no epsr <epsr-instance> enhancedrecovery enable
```

Parameter	Description
<epsr-instance>	Name of the EPSR instance.

Default Default is enhancedrecovery mode disabled.

Mode EPSR Configuration

Example To apply enhanced recovery on the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue enhancedrecovery enable
```

Related Commands **show epsr**

epsr mode master controlvlan primary port

This command creates a master EPSR instance.

Note Running your switch as an EPSR master node requires a Premium License.

Syntax

```
epsr <epsr-instance> mode master controlvlan <2-4094> primaryport
   <port>
```

Parameter	Description
<epsr-instance>	Name of the EPSR instance.
`mode`	Determines the node is acting as a master.
`master`	Sets switch to be the master node for the named EPSR ring.
`controlvlan`	The VLAN that will transmit EPSR control frames.
<2-4094>	VLAN id.
`primaryport`	Primary port for the EPSR instance.
<port>	The primary port. The port may be a switch port (e.g. `port1.0.4`) or a static channel group (e.g. `sa2`). It cannot be a dynamic (LACP) channel group.

Note The software allows you to configure more than two ports or static channel groups to the control VLAN within a single switch or stacked node. However, we advise against this because in certain situations it can produce unpredictable results.

If the control VLAN contains more than two ports (or static channels) an algorithm selects the two ports or channels with the lowest number to be the ring ports. However if the switch has only one channel group is defined to the control vlan, EPSR will not operate on the secondary port.

EPSR does not support Dynamic link aggregation (LACP).

Mode EPSR Configuration

Example To create a master EPSR instance called `blue` with `vlan2` as the control VLAN and `port1.0.1` as the primary port, use the command:

```
awplus(config-epsr)# epsr blue mode master controlvlan vlan2
                     primaryport port1.0.1
```

Related Commands **epsr mode transit controlvlan**
show epsr

epsr mode transit controlvlan

This command creates a transit EPSR instance.

Syntax `epsr <epsr-instance> mode transit controlvlan <2-4094>`

Parameter	Description
<epsr-instance>	Name of the EPSR instance.
`mode`	Determines the node is acting as a transit node.
`transit`	Sets switch to be the transit node for the named EPSR ring.
`controlvlan`	The VLAN that will transmit EPSR control frames.
<2-4094>	VLAN id.

Note

The software allows you to configure more than two ports or static channel groups to the control VLAN within a single switch or stacked node. However, we advise against this because in certain situations it can produce unpredictable results.

If the control VLAN contains more than two ports (or static channels) an algorithm selects the two ports or channels with the lowest number to be the ring ports. However if the switch has only one channel group is defined to the control vlan, EPSR will not operate on the secondary port.

EPSR does not support Dynamic link aggregation (LACP).

Mode EPSR Configuration

Example To create a transit EPSR instance called `blue` with `vlan2` as the control VLAN, use the command:

```
awplus(config-epsr)# epsr blue mode transit controlvlan vlan2
```

Related Commands

epsr mode master controlvlan primary port
epsr mode transit controlvlan
show epsr

epsr priority

This command sets the priority of an EPSR instance on an EPSR node. Priority is used to prevent "superloops" forming under fault conditions with particular ring configurations. Setting a node to have a priority greater than one, also has the effect of turning on **superloop protection**.

The **no** variant of this command returns the priority of the EPSR instance back to its default value of 0, which also disables EPSR Superloop prevention.

Syntax

```
epsr <epsr-instance> priority <0-127>

no <epsr-instance> priority
```

Parameter	Description
<epsr-instance>	Name of the EPSR instance.
`priority`	The priority of the ring instance selected by the epsr-name parameter.
<0-127>	The priority to be applied (0 is the lowest priority and represents no superloop protection).

Default The default priority of an EPSR instance on an EPSR node is 0. The negated form of this command resets the priority of an EPSR instance on an EPSR node to the default value.

Mode EPSR Configuration

Example To set the priority of the EPSR instance called `blue` to the highest priority (127), use the command:

```
awplus(config-epsr)# epsr blue priority 127
```

To reset the priority of the EPSR instance called `blue` to the default (0), use the command:

```
awplus(config-epsr)# no epsr blue priority
```

Related Commands **epsr configuration**

epsr state

This command enables or disables an EPSR instance.

Syntax `epsr <epsr-instance> state {enabled|disabled}`

Parameter	Description
<epsr-instance>	The name of the EPSR instance.
`state`	The operational state of the ring.
`enabled`	EPSR instance is enabled.
`disabled`	EPSR instance is disabled.

Mode EPSR Configuration

Example To enable the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue state enabled
```

Related Commands **epsr mode master controlvlan primary port**
epsr mode transit controlvlan

epsr trap

This command enables SNMP traps for an EPSR instance. The traps will be sent when the EPSR instance changes state.

The **no** variant of this command disables SNMP traps for an EPSR instance. The traps will no longer be sent when the EPSR instance changes state.

Syntax

```
epsr <epsr-instance> trap
no epsr <epsr-instance> trap
```

Parameter	Description
<epsr-instance>	Name of the EPSR instance.
`trap`	SNMP trap for the EPSR instance.

Mode EPSR Configuration

Example To enable traps for the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# epsr blue trap
```

To disable traps for the EPSR instance called `blue`, use the command:

```
awplus(config-epsr)# no epsr blue trap
```

Related Commands
epsr mode master controlvlan primary port
epsr mode transit controlvlan
show epsr

show debugging epsr

This command shows the debugging modes enabled for EPSR.

Syntax

```
show debugging epsr
```

Mode User Exec and Privileged Exec

Example To show the enabled debugging modes, use the command:

```
awplus# show debugging epsr
```

Related Commands
debug epsr

show epsr

This command displays information about all EPSR instances.

Syntax `show epsr`

Mode User Exec and Privileged Exec

Example To show the current settings of all EPSR instances, use the command:

```
awplus# show epsr
```

Output The following examples show the output display for a **non** superloop topology network.

Figure 84-1: Example output from the show epsr command run on a transit node

```
EPSR Information
------------------------------------------------------------------
 Name ......................... test2
 Mode ......................... Transit
 Status ....................... Enabled
 State ........................ Links-Up
 Control Vlan ................. 2
 Data VLAN(s) ................. 10
 Interface Mode ............... Ports Only
 First Port ................... port1.0.1
 First Port Status ............ Down
 First Port Direction ......... Unknown
 Second Port .................. port1.0.2
 Second Port Status ........... Down
 Second Port Direction ........ Unknown
 Trap ......................... Enabled
 Master Node .................. Unknown
 Enhanced Recovery ............ Disabled
------------------------------------------------------------------
```

Figure 84-2: Example output from the show epsr command run on a master node

```
EPSR Information
------------------------------------------------------------------
 Name ......................... test4
 Mode ......................... Master
 Status ....................... Enabled
 State ........................ Complete
 Control Vlan ................. 4
 Data VLAN(s) ................. 20
 Interface Mode ............... Ports Only
 Primary Port ................. port1.0.3
 Primary Port Status .......... Forwarding
 Secondary Port ............... port1.0.4
 Secondary Port Status ........ Forwarding
 Hello Time ................... 1 s
 Failover Time ................ 2 s
 Ring Flap Time ............... 0 s
 Trap ......................... Enabled
 Enhanced Recovery ............ Disabled
------------------------------------------------------------------
```

Note

The above screen is only viewable when running the switch as an EPSR Master. Running the switch as a master requires a Premium license.

The following examples show the output display for superloop topology network.

Figure 84-3: Example output from the show epsr command run on a Master Node

```
 EPSR Information
----------------------------------------------------------------
 Name ....................... test4
 Mode ........................ Master
 Status ...................... Enabled
 State ....................... Complete
 Control Vlan ................ 4
 Data VLAN(s) ................ 20
 Interface Mode .............. Ports Only
 Primary Port ................ port1.0.3
   Status .................... Forwarding (logically blocking)
   Is On Common Segment ...... No
   Blocking Control .......... Physical
 Secondary Port .............. port1.0.4
   Status .................... Blocked
   Is On Common Segment ...... No
   Blocking Control .......... Physical
 Hello Time .................. 1 s
 Failover Time ............... 2 s
 Ring Flap Time .............. 0 s
 Trap ........................ Enabled
 Enhanced Recovery ........... Disabled
 SLP Priority ................ 12
----------------------------------------------------------------
```

Note The above screen is only viewable when running the switch as an EPSR Master. Running the switch as a master requires a Premium license.

Figure 84-4: Example output from the show epsr command run on a Transit Node

```
 EPSR Information
----------------------------------------------------------------
 Name ....................... test4
 Mode ........................ Transit
 Status ...................... Enabled
 State ....................... Complete
 Control Vlan ................ 4
 Data VLAN(s) ................ 20
 Interface Mode .............. Ports Only
 Primary Port ................ port1.0.3
   Status .................... Forwarding (logically blocking)
   Is On Common Segment ...... No
   Blocking Control .......... Physical
 Secondary Port .............. port1.0.4
   Status .................... Blocked
   Is On Common Segment ...... No
   Blocking Control .......... Physical
 Hello Time .................. 1 s
 Failover Time ............... 2 s
 Ring Flap Time .............. 0 s
 Trap ........................ Enabled
 Enhanced Recovery ........... Disabled
 SLP Priority ................ 12
----------------------------------------------------------------
```

Table 84-1: Parameters displayed in the output of the show epsr command

Parameter on Master Node	Parameter on Transit Node	Description
Name	Name	The name of the EPSR instance.
Mode	Mode	The mode in which the EPSR instance is configured - either Master or Transit
Status	Status	Indicates whether the EPSR instance is enabled or disabled
State	State	Indicates state of the EPSR instance's state machine. Master states are: Idle, Complete, and Failed. Transit states are Links-Up, Links-Down, and Pre-Forwarding.
Control Vlan	Control Vlan	Displays the VID of the EPSR instance's control VLAN.
Data VLAN(s)	Data VLAN(s)	The VID(s) of the instance's data VLANs.
Interface Mode	Interface Mode	Whether the EPSR instance's ring ports are both physical ports (Ports Only) or are both static aggregators (Channel Groups Only).
Primary Port	First Port	The EPSR instance's primary ring port.
- Status	- Status	Whether the ring port is forwarding (Forwarding) or blocking (Blocked), or has link down (Down), and if forwarding or blocking, "(logical)" indicates the instance has only logically set the blocking state of the port because it does not have physical control of it.
	- Direction	The ring port on which the last EPSR control packet was received is indicated by "Upstream". The other ring port is then "Downstream"
- Is On Common Segment	- Is On Common Segment	Whether the ring port is on a shared common segment link to another node, and if so, "(highest rank)" indicates it is the highest priority instance on that common segment.
- Blocking Control	- Blocking Control	Whether the instance has "physical" or "logical" control of the ring port's blocking in the instance's data VLANs.
Secondary Port	Second Port	The EPSR instance's secondary port.
- Status	- Status	Whether the ring port is forwarding (Forwarding) or blocking (Blocked), or has link down (Down), and if forwarding or blocking, "(logical)" indicates the instance has only logically set the blocking state of the port, because it does not have physical control of it. Note that on a master configured for SuperLoop Prevention (non-zero priority) its secondary ring port can be physically forwarding, but logically blocking. This situation arises when it is not the highest priority node in the topology (and so does not receive LINKS-DOWN messages upon common segment breaks) and a break on a common segment in its ring is preventing reception of its own health messages.
	- Direction	The ring port on which the last EPSR control packet was received is indicated by "Upstream". The other ring port is then "Downstream"
- Is On Common Segment	- Is On Common Segment	Whether the ring port is on a shared common segment link to another node, and if so, "(highest rank)" indicates it is the highest priority instance on that common segment

Table 84-1: Parameters displayed in the output of the show epsr command (cont.)

Parameter on Master	Parameter on Transit	Description(cont.)
- Blocking Control	- Blocking Control	Whether the instance has “physical” or “logical” control of the ring port's blocking in the instance's data VLANs
Hello Time		The EPSR instance's setting for the interval between transmissions of health check messages (in seconds)
Failover Time		The time (in seconds) the EPSR instance waits to receive a health check message before it decides the ring is down
Ring Flap Time		The minimum time the EPSR instance must remain in the failed state
Trap	Trap	Whether the EPSR instance has EPSR SNMP traps enabled
Enhanced Recovery	Enhanced Recovery	Whether the EPSR instance has enhanced recovery mode enabled
SLP Priority	SLP Priority	The EPSR instance's priority (for SuperLoop Prevention)

Related Commands

epsr mode master controlvlan primary port
epsr mode transit controlvlan
show epsr counters

show epsr common segments

This command displays information about all the superloop common segment ports on the switch

Syntax `show epsr common segments`

Example To display information about all the superloop common segment ports on the switch, use the command:

```
awplus# show epsr common segments
```

Figure 84-5: Example output from the show epsr common segments command

```
EPSR Common Segments

Common Seg  EPSR                          Port     Phys Ctrl  Ring
Ring Port   Instance         Mode     Prio  Type     of Port?   Port Status
-------------------------------------------------------------------------------
port1.0.24  test_inst_Red    Transit  127   Second     Yes      Fwding
            test_inst_Blue   Transit  126   Second     No       Fwding (logical)
            test_inst_Green  Transit  125   First      No       Fwding (logical)

sa4         testA            Master    15   Primary    Yes      Blocking
            testB            Transit   14   Second     No       Fwding (logical)

sa5         test_55          Transit    8   First      Yes      Down
            test_77          Transit    7   First      No       Down
-------------------------------------------------------------------------------
```

Related Commands
- show epsr
- show epsr summary
- show epsr counters

show epsr config-check

This command checks the configuration of a specified EPSR instance, or all EPSR instances.

If an instance is enabled, this command will check for the following errors or warnings:

- The control VLAN has the wrong number of ports.
- There are no data VLANs.
- Some of the data VLANs are not assigned to the ring ports.
- The failover time is less than 5 seconds, for a stacked device.
- The instance is a master that shares a common segment with a higher priority instance.
- The instance is a master that shares a common segment with another master.
- The instance is a master with its secondary port on a common segment.Syntax.

Syntax `show epsr <instance> config-check`

Parameter	Description
<instance>	Name of the EPSR instance to check on.

Mode User Exec and Privileged Exec

Example To check the configuration of all EPSR instances and display the results, use the command:

```
awplus# show epsr config-check
```

Figure 84-6: Example output from the show epsr config-check command

```
EPSR              Status   Description
Instance
------------------------------------------------------------------------------
red               Warning  Failover time is 2s but should be 5s because device.
                             is stacked
white             OK.
blue              Warning  Primary port is not in data VLANs 29-99.
orange            OK.

Don't forget to check that this node's configuration is consistant with all
other nodes in the ring.
------------------------------------------------------------------------------
```

Related Commands show epsr

show epsr <epsr-instance>

This command displays information about the specified EPSR instance.

Syntax `show epsr <epsr-instance>`

Parameter	Description
<epsr-instance>	Name of the EPSR instance.

Mode User Exec and Privileged Exec

Example To show the current settings of the EPSR instance called `blue`, use the command:

```
awplus# show epsr blue
```

Related Commands **epsr mode master controlvlan primary port**
epsr mode transit controlvlan
show epsr counters

show <epsr-instance> counters

This command displays counter information about the specified EPSR instance.

Syntax `show epsr <epsr-instance> counters`

Parameter	Description
<epsr-instance>	Name of the EPSR instance.

Mode User Exec and Privileged Exec

Example To show the counters of the EPSR instance called `blue`, use the command:

```
awplus# show epsr blue counters
```

Related Commands **epsr mode master controlvlan primary port**
epsr mode transit controlvlan
show epsr

show epsr counters

This command displays counter information about all EPSR instances.

Syntax `show epsr counters`

Mode User Exec and Privileged Exec

Example To show the counters of all EPSR instances, use the command:

```
awplus# show epsr counters
```

Related Commands
epsr mode master controlvlan primary port
epsr mode transit controlvlan
show epsr

show epsr summary

This command displays summary information about all EPSR instances on the switch

Syntax `show epsr summary`

Mode User Exec and Privileged Exec

Example To display epsr summary information, use the command:

```
awplus# show epsr summary
```

Example To display EPSR summary information, use the command:

```
awplus# show epsr summary
```

Figure 84-7: Example output from the show epsr summary command

```
EPSR Summary Information

Abbreviations:
  M = Master node
  T = Transit node
  C = is on a common segment with other instances
  P = instance on a common segment has physical control of the shared port's
        data VLAN blocking
  LB = ring port is Logically Blocking - applicable to master only

EPSR                                    Ctrl       Primary/1st     Secondary/2nd
Instance      Mode Status   State       VLAN  Prio Port Status     Port Status
-------------------------------------------------------------------------------
test-12345     T  Enabled  Links-Down     6  127  Blocking (C,P)  Blocking (C,P)
test1          M  Enabled  Complete       5   12  Fwding          Fwding (LB)
test2          T  Enabled  Pre-Fwding     4  126  Fwding (C)      Blocking (C)
localB         T  Disabled Idle          40    0  Unknown         Unknown
localC         T  Disabled Idle          41    0  Unknown         Unknown
-------------------------------------------------------------------------------
```

undebug epsr

This command applies the functionality of the **no debug epsr** command on page 84.3.

Part 7: Network Management

- Chapter 85 AMF Introduction and Configuration
- Chapter 86 AMF Commands
- Chapter 87 NTP Introduction and Configuration
- Chapter 88 NTP Commands
- Chapter 89 Dynamic Host Configuration Protocol (DHCP) Introduction
- Chapter 90 Dynamic Host Configuration Protocol (DHCP) Commands
- Chapter 91 DHCP for IPv6 (DHCPv6) Introduction and Configuration
- Chapter 92 DHCP for IPv6 (DHCPv6) Commands
- Chapter 93 SNMP Introduction
- Chapter 94 SNMP Commands
- Chapter 95 SNMP MIBs
- Chapter 96 LLDP, LLDP-MED and Voice VLAN Introduction and Configuration
- Chapter 97 LLDP Commands
- Chapter 98 SMTP Commands
- Chapter 99 RMON Introduction and Configuration
- Chapter 100 RMON Commands
- Chapter 101 Triggers Introduction
- Chapter 102 Triggers Configuration

- Chapter 103 Trigger Commands
- Chapter 104 Ping Polling Introduction and Configuration
- Chapter 105 Ping-Polling Commands
- Chapter 106 sFlow Introduction and Configuration
- Chapter 107 sFlow Commands
- Chapter 108 Cable Fault Locator Introduction
- Chapter 109 Cable Fault Locator Commands

Chapter 85: AMF Introduction and Configuration

Introduction to AMF 85.2
AMF Supported Products and Software Versions 85.2
Key Benefits of AMF 85.3
- Unified Command-Line 85.3
- Configuration Backup and Recovery 85.3
- Rolling-Reboot Upgrade 85.3
- Node Provisioning 85.4

AMF Terminology and Introduction 85.5
- AMF Network 85.5
- AMF Nodes 85.5
- Node Licensing 85.5
- Node Interconnection 85.6
- AMF Domains 85.6

AMF Network Operational Concepts 85.8
- Retention and Use of the 'Manager' Username 85.8
- Working-Set 85.8
- AMF Restricted Login 85.9
- Loop-Free Data Plane 85.9
- Aggregators 85.9
- VCStacks 85.9
- AMF External Removable Media 85.9
- AMF Interaction with QoS and ACLs 85.10
- NTP and AMF 85.10

Configuring AMF 85.12
- AMF Tunneling (Virtual Links) 85.18

Verifying the AMF Network 85.22
Configuring Multiple Nodes at the Same Time: the Unified CLI 85.24
- Working-Set Groups 85.25
- Executing Commands on Working-Sets 85.27
- Interactive Commands 85.29

AMF Backups 85.30
- Using External Media Storage 85.30
- Performing a Manual Backup 85.31
- Backing up to Remote Servers 85.32

Node Recovery 85.34
- Automatic Node Recovery 85.34
- Restoring a Node to a "Clean" State 85.36
- Manual Node Recovery 85.37
- Node Recovery on VCStacks 85.38

AMF Safe Configuration 85.39
- Detecting AMF Safe Configuration Operation 85.39
- AMF Safe Configuration Procedures 85.39
- Undoing an AMF Safe Configuration 85.40

Rolling-Reboot Firmware Upgrade 85.42
- Performing a Rolling-Reboot Upgrade 85.44

Introduction to AMF

The Allied Telesis Management Framework (AMF) is a suite of features that combine to simplify network management across all supported network switches from the core to the edge.

AMF also provides simplified switch recovery and firmware upgrade management. The primary function of AMF is to reduce the management and maintenance overhead on a network, while improving on responsiveness and handling of switch failures within the network.

This chapter provides a conceptual introduction to AMF together with its benefits, together with configuration guidelines showing how to use AMF in practical networks. For more information on the commands used in this chapter, see **"AMF Commands" on page 86.1**.

AMF Supported Products and Software Versions

The following list shows which Allied Telesis switches are capable of running AMF and indicates those capable of operating as Master Nodes.

An AMF-Master feature license is required for each AMF master node in the AMF network. AMF-Master feature licenses are available for the SBx8100 and SBx908 platforms.

Table 85-1: AMF Nodal Capability by Switch Type

Switch Type	AMF Nodal Capability
SwitchBlade™ x8100	master or member
SwitchBlade™ x908	master or member
x900 series switches	member only
x610 series switches	member only
x510 series switches	member only
IX5-28GPX switches	member only
x310 series switches	member only
x230 series switches	member only
x210 series switches	member only
DC2552XS switch	member only

Key Benefits of AMF

The key benefits of AMF include its unified command-line, simple configuration backup and recovery process, and time-saving rolling firmware upgrade.

Unified Command-Line

The conventional means of configuring and controlling AlliedWare Plus (AW+) switches is to use their text-based command-line interface (CLI). In existing networks, the CLI is available via a serial console port and also to remote login sessions such as SSH.

AMF extends this capability from managing either a single switch to managing a whole network by using a single (unified) CLI session. Using the unified CLI, a network administrator can nominate all nodes or a subset of nodes within the AMF network to comprise an entity known as a "**working-set**". Commands can then be executed concurrently across all switching nodes within the defined working-set as if they were a single unit. Any existing configuration or diagnostic actions can thus be applied to multiple devices using a single command sequence, thus reducing maintenance costs and configuration complexity, while still retaining complete flexibility in network design and control.

Multiple AMF networks can exist side by side across a single physical network. Note that AMF treats a Virtual Chassis Stack (VCStack) as a single node.

Configuration Backup and Recovery

The **master** nodes use external storage to automatically backup the complete configuration information for all their member nodes, including boot configuration, firmware, licenses, and user scripts.

If an AMF member node should fail, the AMF process will automatically recognize and reconfigure an unconfigured replacement (standby) unit, completely recreating the stored configuration of the failed unit into the replacement unit. The new unit will then reboot and resume service, without any need for user intervention beyond physical hardware replacement and cable connection. In this way AMF provides a complete zero-touch recovery solution. For more information, see **"AMF Backups" on page 85.30**.

Rolling-Reboot Upgrade

Installing Firmware upgrades on a production network is typically an infrequent but sensitive and labor-intensive process. AMF is able to roll-out upgrades to a user-selected subset of nodes. All that needs to be entered is the target group of nodes, and the location where the new firmware is stored; AMF will then take care of the rest. Nodes are upgraded in a serial fashion, with each node tested before continuing the upgrade on the next node.

If an upgrade fails on a particular node, the upgrade process is automatically terminated and that node will revert to its previous firmware version. In this way firmware updates are almost completely hands-free, whilst also providing confidence that a bad update will not result in loss of service. For more information, see **"Performing a Rolling-Reboot Upgrade" on page 85.44**.

Node Provisioning

It is generally undesirable to have unconfigured devices connected to the network. Node provisioning enables you to preconfigure a port ready to accept and automatically configure a “clean” (as new) device for connection at a later date. This is achieved by storing the future node's configuration in the master node's backup files ready to be loaded to the new device when connected.

AMF Terminology and Introduction

This section contains a glossary of terminology used to describe AMF networking.

AMF Network

Conceptually an AMF network is a collection of interconnected network switch nodes. This interconnection in turn comprises a hierarchy of network domains. These terms are explained in more detail later in this chapter.

Network name

Because networks are able to interconnect, an AMF network *name* is necessary to identify the AMF network to which any given node belongs. It follows therefore, that all nodes within a single AMF network must be configured with the same AMF network name.

AMF Nodes

Two types of nodes exist within an AMF network, Master Nodes and Member Nodes. Either type can comprise either a single switch, or a VCStack.

Master Nodes

Master nodes are user defined by configuration. They then form the core domain of the AMF network. Aspects of master node functionality include:

- performing file system backups of all nodes in the AMF network.
- providing an essential component for the formation of an AMF network. That is, an AMF network cannot exist without the existence of at least one master node.
- at least one master node must be present for an AMF network to exist.

When more than one AMF master node exists in an AMF network, their operation is completely independent and unsynchronized.

Member Nodes

AMF member nodes are referred to simply as nodes.

Node Licensing

Master node License

AMF master nodes are supported on selected switch platforms: an AMF license is required for each master. For a list of node capability against specific switch types, see **Table 85-1 on page 85.2**

A VCStack needs to have consistent licensing on all stack members. Therefore, an AMF master license would be required on both devices in an SBx908 stack.

When more than one AMF master node exists in an AMF network, it is important to know that these operate completely independently of each other, and there is no synchronization between AMF master nodes.

For redundancy, an AMF network can have multiple master nodes, each acting as a master for the network. However, there is no synchronization of status or data files between the masters. The behavior of a master node is not changed at all by the presence of other master nodes.

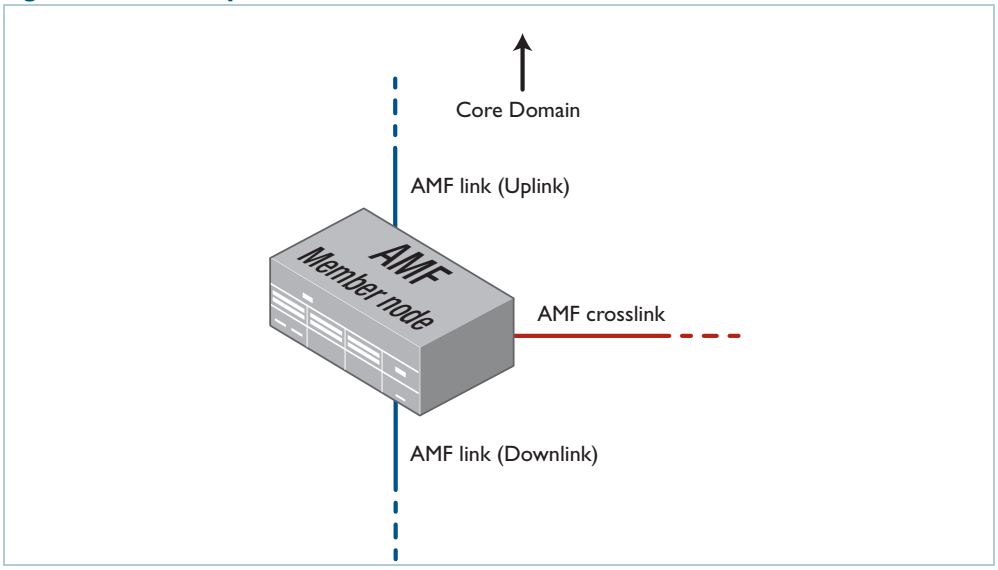
Core distance

This is the distance (hop count) between a particular domain and its Core domain. The Core domain has a Core distance of 0, and the maximum recommended Core distance in an AMF network is 8.

Node Interconnection

Nodes can connect either horizontally using crosslinks, or vertically using Uplinks/ Downlinks. This is shown in the illustration below:

Figure 85-1: AMF Uplinks, Downlinks, and Crosslinks



AMF links, of either type, are used to pass AMF management traffic between nodes; however, they can also carry other network traffic. Configuring an interface as an **AMF-link** will automatically put the port into trunk mode. An AMF link can be either a single link or a static aggregator. For more information on trunk mode see **"Configuring VLANs" on page 18.3**.

Crosslinks AMF crosslinks are used to connect AMF nodes to other AMF nodes within what is termed an AMF Domain. Configuring an interface as an AMF-crosslink will automatically put its port into trunk mode. A crosslink can be either a single link or a static aggregator.

AMF master nodes must be connected using AMF crosslinks to ensure they are part of the uppermost domain level.

Up/Down Links Uplinks/Downlinks interconnect domains in what is a vertical hierarchy, the highest domain being the core domain.

AMF Domains

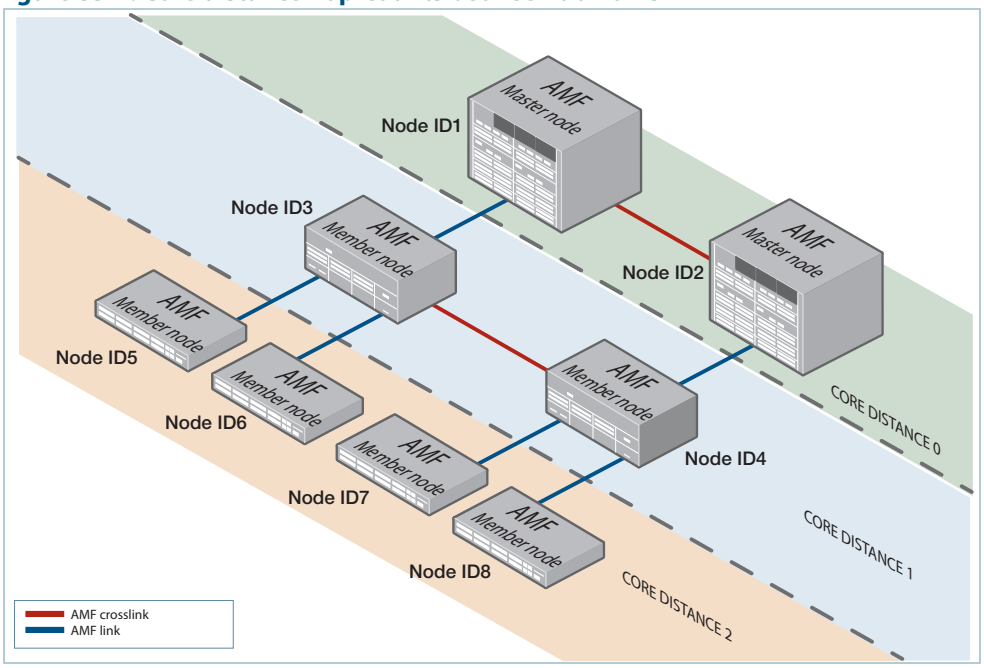
Every AMF node belongs to an AMF domain. Domains can comprise of a single node or multiple nodes. AMF master nodes are included in highest domain level, sometimes called the core domain, and all other domains are rooted in this domain.

As previously mentioned, AMF domains are determined by AMF crosslinks, (see **"Working-Set" on page 85.8**). All nodes connected via AMF crosslinks form part of the same domain, and nodes connected via regular AMF links will be part of either higher of lower level domains.

Nodes within a domain must be connected in either a chain or ring topology. This means that a maximum of two crosslinks should be configured on any single node. The advantage of an AMF domain is that two links from a domain to a single higher level domain will provide redundant AMF links. We recommend that an AMF domain should only be connected to a single higher level domain, though it may be connected to multiple lower level domains. We also recommend that you set a maximum number of 12 nodes per domain.

Hop-Count The vertical distance of a domain from the core domain is known as its Hop-Count. The illustration**"Core distance hop-counts between domains" on page 85.7** shows the relationship between nodes, domains and core distance (hop-count).

Figure 85-2: Core distance hop-counts between domains



Node provisioning Node provisioning enables you to configure a node before it is physically present in the AMF network. When the node is eventually connected to an expectant port, it will automatically set itself up with the previously stored configuration files and release.

AMF Network Operational Concepts

Retention and Use of the 'Manager' Username

The default **username** for an AlliedWare Plus login is "manager", with a documented default **password**. Users should change this password on all their nodes to provide login security.

It is possible to add new usernames and passwords to nodes, but to retain the ability to centrally manage the network, usernames should be uniformly configured across all AMF nodes within the AMF network.

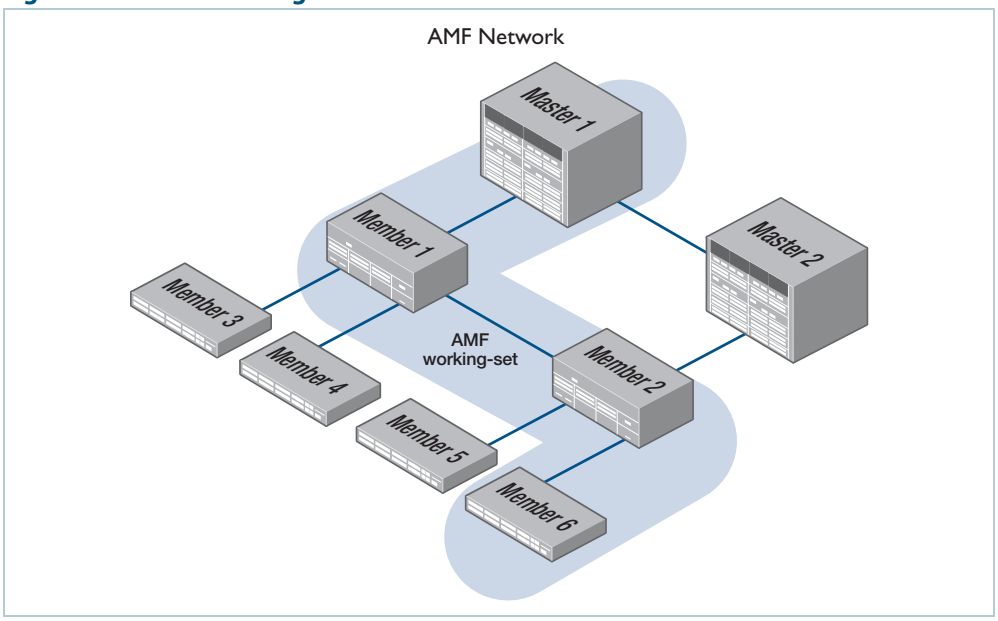
Note that managing a network with AMF is incompatible with user authentication via RADIUS or TACACS+. Use the normal local database for user authentication.

Working-Set

Conceptually a working set is a collection of switches that can then be configured centrally as if there were a single device. A working set may comprise a pre-defined group that has been automatically created based on some common set of physical attributes such as switch type etc, or it may be created by a network user for ease of configuration.

Specifying or selecting a working-set allows CLI commands to be executed on all nodes within the selected working-set with a single command. A working-set can be defined, selected and configured from any node within an AMF network. The illustration**"AMF working-set" on page 85.8** shows a number of switches which comprise a working set.

Figure 85-3: AMF working-set



AMF Restricted Login

By default, a user logged into any node on an AMF network is able to manage any other node by using either working-sets or AMF remote login (provided the login username exists on all nodes). Where the access provided by this feature is too wide, or contravenes network security restrictions, this access can be limited by running the command **"atmf restricted-login" on page 86.46**. This command will not be saved in the running configuration; it is a network property that can be enabled or disabled from any AMF master. The status of restricted-login will be retained over a reboot.

When restricted login is enable on the network, only the AMF Master nodes are able to create working-sets or manage other devices via AMF remote-logins. Other nodes may remote login to the AMF Master, but they will require password authentication on that master, and will then be able to create working-sets originating from the Master.

Note that once you have run this command, certain other commands that utilize the AMF working-set command, such as the **include**, **atmf reboot-rolling** and **show atmf group members** commands, will operate only on master nodes.

Loop-Free Data Plane

The current version of AMF does not control the data plane, so it is a requirement that the network is configured such that the data plane (i.e. the paths defined by the data VLANs) is kept loop free.

Aggregators

Dynamic Aggregators (LACP) cannot be used on ports configured as AMF links or cross-links. Therefore any aggregated links in an AMF network need to be configured as static aggregators.

VCStacks

If any VCStacks are included as AMF nodes it is a requirement that the *VCS virtual MAC* feature is enabled to ensure correct operation of the AMF network. If the VCStack is running as an AMF master node and backup is required, then removable external storage media should be installed in both stack members.

AMF External Removable Media

In order to maintain a recovery capability, all AMF master nodes require external storage media installed, such as a USB or SD card. This external storage is used to hold a backup of all relevant files from all nodes within the AMF network, including other master nodes, so it must be large enough to accommodate all of the backed up files. Files that are backed up include all configuration files, release files, and scripts, but do not include files like core dumps, exception logs, or technical support files.

Typically a *4GB* capacity external media device would be of sufficient size to hold backups for a *40 node* AMF network.

AMF Interaction with QoS and ACLs

It's important that ACL and QoS rules do not block any traffic on VLANs 4091 and 4092 because they are the default AMF control VLANs. Similarly, ACL and QoS rules should not block any Layer 3 traffic on 172.31.0.* or 172.31.128.* these being the default AMF management traffic subnets. Packets with protocol type 0xfbae and BPDU packets that use the MAC address: 0180.c200.002e should also not be blocked.

Note The AMF control VLANs and AMF management subnets can be manually changed.

With AMF enabled, the number of ACLs on the DC2552XS, x210, 230, x310, x510 and x610 switches decreases from 2049 to 2048. If this is problematic and you are not using AMF, you can disable AMF, which will allow the previous maximum of 2049.

NTP and AMF

AMF uses NTP to synchronize the system clocks across nodes within the network. For this to operate, one or more external NTP servers must be configured on the network, and every node on the network must be configured to use the external server or servers.

Alternatively, you can configure an AlliedWare Plus device as the NTP master, but this NTP master must not be a member of the AMF network. Otherwise, NTP synchronization issues can occur.

To configure an AlliedWare Plus device as an NTP master, use the command:

```
awplus(config)# ntp master 11
```

The primary function of NTP within an AMF network is to ensure that time and date stamps on backups are consistent across member nodes within the backup. This is particularly important in an AMF network that has multiple AMF master nodes, to ensure that node recovery is performed with the most up-to-date backup.

Configuring NTP on the AMF network

Before you configure NTP on the AMF network, we recommend setting all nodes in the network to the same time, date, and timezone, to ensure NTP synchronization. To do this, create an AMF working-set of the whole network and set the date and time, for example:

```
awplus(config)# atmf working-set group all
awplus(config)# clock set 16:47:00 11 Sep 2014
awplus(config)# clock timezone utc plus 12
```

Once you have configured all nodes with the same time, date and timezone, configure the working-set of all nodes with the IP address of the NTP server, for example:

```
awplus(config)# ntp server 172.31.0.1
```

You can then check that the nodes have synchronized with the NTP server using the **show ntp status** command, for example:

```
awplus# show ntp status
```

```
awplus#show ntp status
Clock is synchronized, stratum 13, reference is 172.31.0.1
actual frequency is 7.1420 PPM, precision is 2**-18
reference time is d7bba834.19f1a68f (16:48:52.101 utc Thu Sep 11
2014)
clock offset is -1.286 msec, root delay is 2.237 msec
root dispersion is 45.213 msec
```

Configuring AMF

The following configuration example uses a simplified network to explain the steps required to configure AMF.

Figure 85-4: Simple AMF single master example

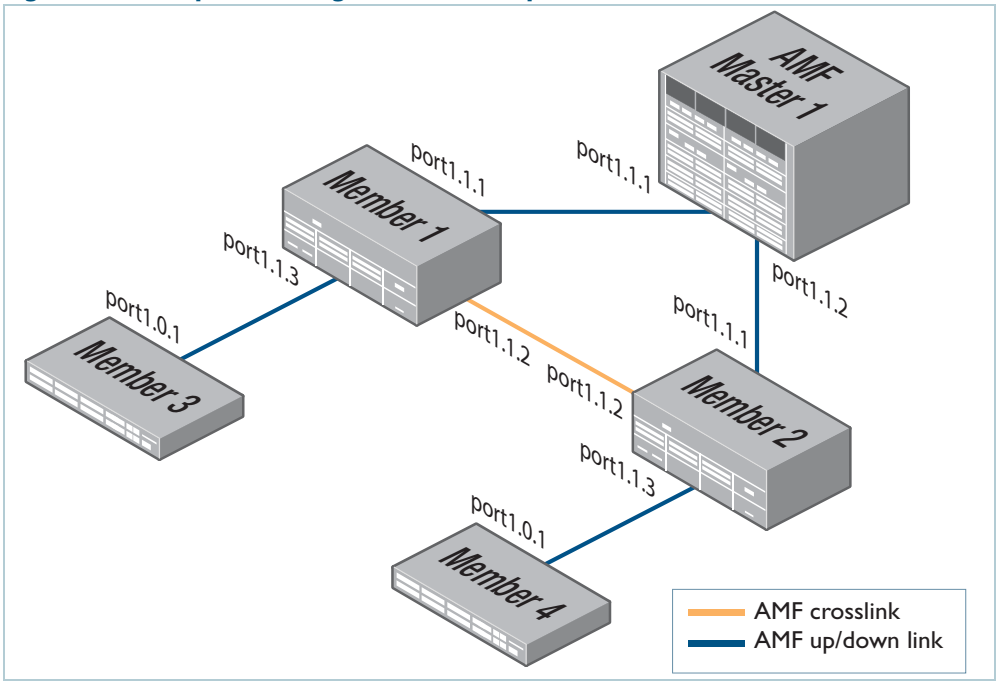


Table 85-2: Configure the AMF master node

Description	Prompt	Command
Step 1. Set the host name.		
Enter Global Configuration mode	`(awplus#)`	`configure terminal`
Set the host name	`awplus(config)#)`	`hostname AMF_Master`
Note that host names are used as the AMF node name and **MUST BE UNIQUE** within the AMF network.		
Step 2. Set the AMF network name.		
Set the AMF network name.	`AMF_Master(config)#`	`atmf network-name atmf1`
Note that the AMF network name must be the same on all nodes within the AMF network, and the device must be *rebooted* before the AMF network name takes effect.		
Step 3. Configure the switch to be the AMF master.		
	`AMF_Master(config)#`	`atmf master`
An AMF network must have at least one master configured. A license is required for each AMF master in the AMF network. If an AT-SBx8100 with dual CFCs is configured as an AMF master a license is only required on the CFC master, as the license with be synchronized across CFCs. If an AT-SBx908 is configured as an AMF master, a license is required to be installed on **both** stack members.		
Step 4. Configure the data VLANs.		
	`AMF_Master(config)#`	`vlan database`
	`AMF_Master(config-vlan)#`	`vlan 2-3`
Step 5. Configure ports as AMF-links.		
	`AMF_Master(config)#`	`interface port1.1.1-1.1.2`
	`AMF_Master(config-if)#`	`switchport atmf-link`
Step 6. Configure data VLANs on AMF-links as required.		
	`AMF_Master(config-if)#`	`switchport trunk allowed vlan add 2-3`
Step 7. Save the configuration and reboot the switch.		
	`AMF_Master#`	`copy running-config startup-config`
Building configuration...[OK]		
	`AMF_Master#`	`reload`
Are you sure you want to reboot the whole chassis? (y/n): y		

Table 85-3: Configure the first member node (Member1)

Description	Prompt	Command
Step 1. Set the host name.		
Enter Global Configuration mode	`(awplus#)`	`configure terminal`
Set the host name	`awplus(config)#)`	`hostname Member1`
Note that host names are used as the AMF node name and **MUST BE UNIQUE** within the AMF network.		
Step 2. Set the AMF network name.		
Set the AMF network name to atmf1.	`Member1(config)#`	`atmf network-name atmf1`
Note that the AMF network name must be the same on all nodes within the AMF network, and the device must be *rebooted* before the AMF network name takes effect.		
Step 3. Configure the data VLANs.		
Enter the VLAN Configuration mode	`Member1(config)#`	`vlan database`
Create VLANs 2 and 3	`Member1(config-vlan)#`	`vlan 2-3`
Step 4. Configure ports as AMF-links.		
Enter Interface Configuration mode for ports 1.1.1 to 1.1.3	`Member1(config)#`	`interface port1.1.1-1.1.3`
Configure these ports as AMF links	`Member1(config-if)#`	`switchport atmf-link`
Step 5. Configure data VLANs on the AMF-links as required.		
Set VLANs 2 to 3 to be data VLANs	`Member1(config-if)#`	`switchport trunk allowed vlan add 2-3`
Step 6. Configure AMF-crosslink.		
Enter the Interface Configuration mode for port 1.1.2	`Member1(config)#`	`interface port1.1.2`
Set this port to be an AMF-crosslink	`Member1(config-if)#`	`switchport atmf-crosslink`
	`Member1(config-if)#`	`switchport trunk native vlan none`
Note that AMF links and crosslinks do not need to be configured with data VLANs and can be used solely to provide AMF management VLAN redundancy.		
Step 7. Save the configuration and reboot the switch.		
	`Member1#`	`copy running-config startup-config`
Building configuration...[OK]		
	`Member1#`	`reload`
Are you sure you want to reboot the whole chassis? (y/n):		`y`

Table 85-4: Configure the first member node (Member2)

Description	Prompt	Command
Step 1. Set the host name.		
Enter Global Configuration mode	awplus#)	configure terminal
Set the host name	awplus(config)#)	hostname Member2
Note that host names are used as the AMF node name and **MUST BE UNIQUE** within the AMF network.		
Step 2. Set the AMF network name.		
Set the AMF network name to atmf1.	Member2(config)#	atmf network-name atmf1
Note that the AMF network name must be the same on all nodes within the AMF network, and the device must be *rebooted* before the AMF network name takes effect.		
Step 3. Configure the data VLANs.		
Enter the VLAN Configuration mode	Member2(config)#	vlan database
Create VLANs 2 and 3	Member2(config-vlan)#	vlan 2-3
Step 4. Configure ports as AMF-links.		
Enter Interface Configuration mode for ports 1.1.1 to 1.1.3	Member2(config)#	interface port1.1.1-1.1.3
Configure these ports as AMF links	Member2(config-if)#	switchport atmf-link
Step 5. Configure data VLANs on the AMF-links as required.		
Set VLANs 2 to 3 to be data VLANs	Member2(config-if)#	switchport trunk allowed vlan add 2-3
Step 6. Configure AMF-crosslink.		
Enter the Interface Configuration mode for port 1.1.2	Member2(config)#	interface port1.1.2
Set this port to be an AMF-crosslink	Member2(config-if)#	switchport atmf-crosslink
	Member2(config-if)#	switchport trunk native vlan none
Note that AMF links and crosslinks do not need to be configured with data VLANs and can be used solely to provide AMF management VLAN redundancy.		
Step 7. Save the configuration and reboot the switch.		
	Member2#	copy running-config startup-config
Building configuration...[OK]		
	Member2#	reload
Are you sure you want to reboot the whole chassis? (y/n):		y

Table 85-5: Configure the first member node (Member3)

Description	Prompt	Command
Step 1. Set the host name.		
Enter Global Configuration mode	`(awplus#)`	`configure terminal`
Set the host name	`awplus(config)#`	`hostname Member3`
Note that host names are used as the AMF node name and **MUST BE UNIQUE** within the AMF network.		
Step 2. Set the AMF network name.		
Set the AMF network name to atmf1.	`Member3(config)#`	`atmf network-name atmf1`
Note that the AMF network name must be the same on all nodes within the AMF network, and the device must be *rebooted* before the AMF network name takes effect.		
Step 3. Configure the data VLANs		
Enter the VLAN Configuration mode	`Member3(config)#`	`vlan database`
Create VLANs 2 and 3	`Member3(config-vlan)#`	`vlan 2-3`
Step 4. Configure ports as AMF-links.		
Enter Interface Configuration mode for ports 1.0.1 to 1.0.3	`Member3(config)#`	`interface port1.0.1-1.0.3`
Configure these ports as AMF links	`Member3(config-if)#`	`switchport atmf-link`
Step 5. Configure data VLANs on the AMF-links as required.		
Set VLANs 2 to 3 to be data VLANs	`Member3(config-if)#`	`switchport trunk allowed vlan add 2-3`
Step 6. Configure AMF-crosslink.		
Enter the Interface Configuration mode for port 1.0.2	`Member3(config)#`	`interface port1.0.2`
Set this port to be an AMF crosslink	`Member3(config-if)#`	`switchport atmf-crosslink`
	`Member3(config-if)#`	`switchport trunk native vlan none`
Note that AMF links and crosslinks do not need to be configured with data VLANs and can be used solely to provide AMF management VLAN redundancy.		
Step 7. Save the configuration and reboot the switch.		
	`Member3#`	`copy running-config startup-config`
Building configuration...[OK]		
	`Member3#`	`reload`
Are you sure you want to reboot the whole chassis? (y/n):		`y`

.

Table 85-6: Configure the first member node (Member4)

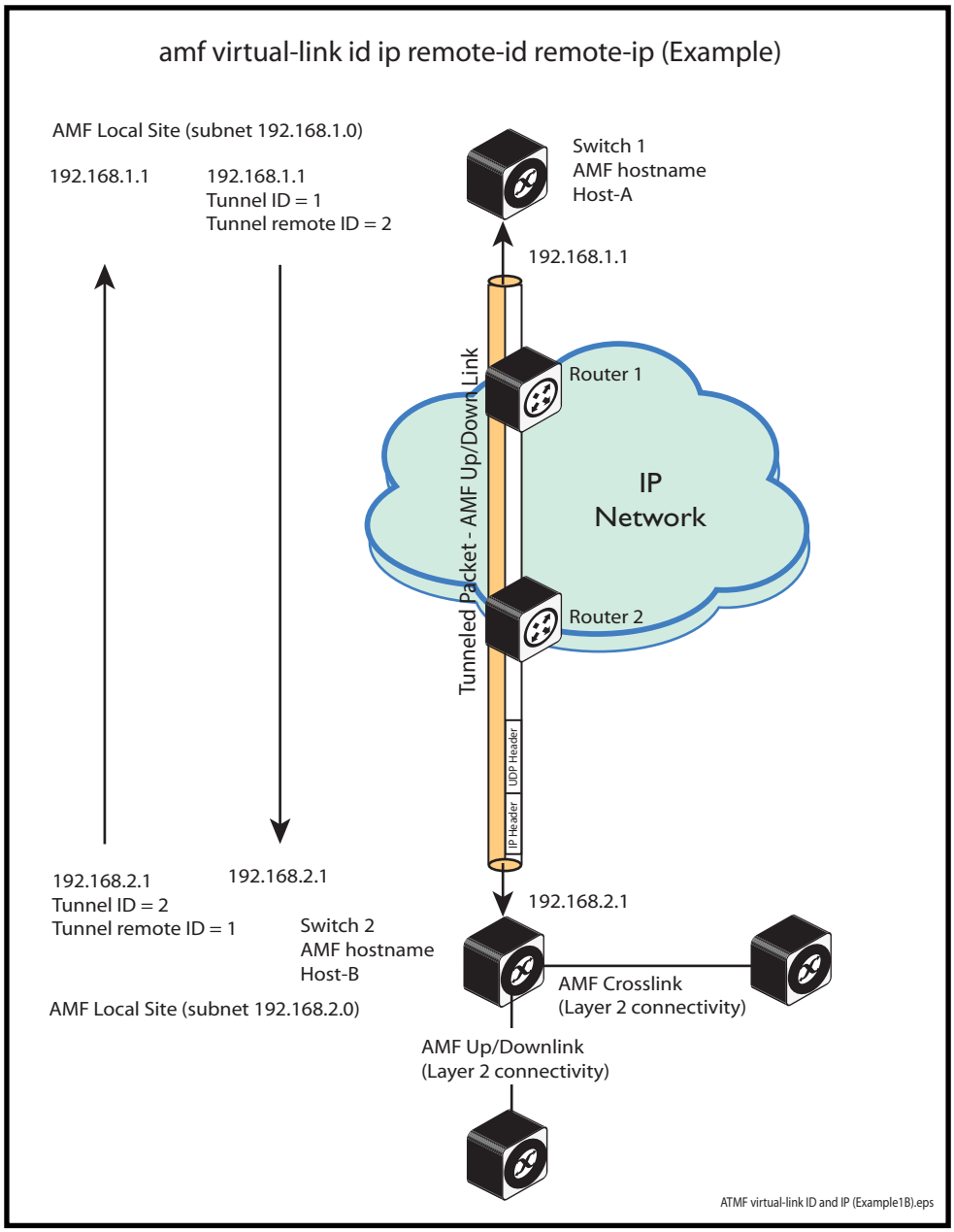
Description	Prompt	Command
Step 1. Set the host name.		
Enter Global Configuration mode	(awplus#)	configure terminal
Set the host name	awplus(config)#)	hostname Member4
Note that host names are used as the AMF node name and **MUST BE UNIQUE** within the AMF network.		
Step 2. Set the AMF network name.		
Set the AMF network name to atmf1.	Member4(config)#	atmf network-name atmf1
Note that the AMF network name must be the same on all nodes within the AMF network, and the device must be *rebooted* before the AMF network name takes effect.		
Step 3. Configure the data VLANs.		
Enter the VLAN Configuration mode	Member4(config)#	vlan database
Create VLANs 2 and 3	Member4(config-vlan)#	vlan 2-3
Step 4. Configure ports as AMF-links.		
Enter Interface Configuration mode for ports 1.0.1 to 1.0.3	Member4(config)#	interface port1.0.1-1.0.3
Configure these ports as AMF links	Member4(config-if)#	switchport atmf-link
Step 5. Configure data VLANs on the AMF-links as required.		
Set VLANs 2 to 3 to be data VLANs	Member4(config-if)#	switchport trunk allowed vlan add 2-3
Step 6. Configure AMF-crosslink.		
Enter the Interface Configuration mode for port 1.0.2	Member4(config)#	interface port1.0.2
Set this port to be an AMF crosslink	Member4(config-if)#	switchport atmf-crosslink
	Member4(config-if)#	switchport trunk native vlan none
Note that AMF links and crosslinks do not need to be configured with data VLANs and can be used solely to provide AMF management VLAN redundancy.		
Step 7. Save the configuration and reboot the switch.		
	Member4#	copy running-config startup-config
Building configuration...[OK]		
	Member4#	reload
Are you sure you want to reboot the whole chassis? (y/n):		y

AMF Tunneling (Virtual Links)

AMF Tunneling enables you to extend your local uplinks and downlinks across a wide area network. The tunneled data is then wrapped in a Layer 3 IP packet for transmission across a wide area IP network. A simple AMF tunnel is shown in **"AMF virtual link" on page 85.18**. Switches 1 and 2 encapsulate the Layer 2 AMF uplink and downlink data and wrap this inside a Layer 3 IP packet to enable it to traverse an IP Network. Routers 1 and 2 (and any other routers within the cloud) perform a conventional routing function, reading the IP addresses of the tunneled packets and forwarding them to their destination.

Once connected through the tunnel, the remote AMF members will have the same AMF capabilities as a directly connected AMF member.

Figure 85-5: AMF virtual link



Configuring a virtual link

The Layer 2 tunnel created by the command **atmf virtual-link id ip remote-id remote-ip** command on page 86.47 enables a local AMF session to appear to pass transparently across a Wide Area Network (WAN) such as the Internet. The addresses configured as the local and remote tunnel IP addresses must have IP connectivity to each other. If the tunnel is configured to connect a head office and branch office over the Internet, typically this would involve using some type of managed WAN service such as a site-to-site VPN. Tunnels are only supported using IPv4.

Configuration involves creating the following:

- local tunnel ID
- local IP address
- remote tunnel ID
- remote IP address

A reciprocal configuration is also required on the corresponding remote device. The local tunnel ID must be unique to the device on which it is configured.

The tunneled link may operate via external (non AlliedWare Plus) routers in order to provide wide area network connectivity. However, in this configuration, these devices perform a conventional router-to-router connection. The protocol tunneling function is accomplished by the AMF nodes.

Note that the requirement to preconfigure the local IP address and tunnel ID on a device located at the far end of an AMF virtual-link tunnel means that zero touch device replacement cannot be achieved on a remote device that terminates the tunnel.

Example Use the following command to create the tunnel shown in figure **Figure 85-5 on page 85.18**.

```
Host-A(config)# atmf virtual-link id 1 ip 192.168.1.1
                remote-id 2 remote-ip 192.168.2.1

Host-B(config)# atmf virtual-link id 2 ip 192.168.2.1
                remote-id 1 remote-ip 192.168.1.1
```

Caution

On an IP interface that is carrying ATMF virtual link traffic, do not set the MTU (Maximum Transmission Unit) below its default value of 1500 bytes.

Prioritizing the tunneled traffic

On the switch that interfaces to the wide area network router, we advise that you prioritize the tunneled traffic directed to the CPU over other CPU-bound user data. You can achieve this by allocating a higher Class of Service (CoS) tag to tunnel traffic than other traffic. The following configuration example shows an appropriate method. In the following configuration example, the virtual link traffic is between IP addresses 192.168.1.1 (on Host-A) and 192.168.2.1 (on Host-B). This connection is mapped to VLAN 10.

Note that the following process will prioritize the AMF traffic only within Switch 1 and Switch 2. To prioritize the tunneled AMF data across the IP network would require applying Layer 3 QoS by applying a DSCP (Differentiated Services Code Point) priority at the network boundary (Router 1 and Router 2) and ensuring that these priority levels are managed throughout the wide area network. Applying and managing QoS through the wide area network is outside the scope of this document.

Virtual link - configuration example

This example is based on the network shown in the illustration **Figure 85-5 on page 85.18**.

In this example, the virtual link traffic flows between 192.168.1.1 at the local end and 192.168.2.1 at the remote end. Subnet 192.168.1.1 exists on vlan10. Note that because this policy is being applied to incoming traffic, the switch IP address should match destination address in the ACL.

Figure 85-6: Prioritizing the tunneled traffic - Configured on Switch 1, Host-A

```
atmf virtual-link id 1 ip 192.168.1.1 remote-id 2 remote-ip
192.168.2.1
[...]

mls qos enable
access-list hardware vlink
 permit ip 192.168.2.1/32 ip 192.168.1.1/32
!
class-map vlink
 match access-group vlink
!
class-map vlinkarp
 match eth-format ethii-any protocol 0806
 match vlan 10
!
policy-map vlink
 class default
 class vlink
  remark new-cos 4 both
 class vlinkarp
  remark new-cos 4 both
!
[...]
!
interface port1.0.10
 switchport
 switchport mode access
 switchport access vlan 10
 service-policy input vlink
!
[...]
interface vlan10
 ip address 192.168.1.1/24
!
```

Table 85-7: Set QoS CoS for an AMF tunneling switch

Description	Prompt	Command
Step 1. Create VLAN 10.		
Enter Global Configuration mode	`(Host-A#)`	`configure terminal`
Enter VLAN config mode	`Host-A(config)#)`	`vlan database`
Create and enable VLAN 10	`Host-A(vlan-config)#`	`vlan 10 name virtual-link state enable`
Step 2. Configure VLAN10		
Enter the VLAN configuration mode for VLAN10.	`Host-A(vlan-config)#`	`interface vlan10`

Table 85-7: Set QoS CoS for an AMF tunneling switch(cont.)

Description	Prompt	Command
Set the IP address for VLAN10 to be 192.168.1.1/24	`Host-A(vlan-config-if)#`	`ip address 192.168.1.1/24`
Return to config mode	`Host-A(config-if)#`	`exit`
Step 3. Add policy map vlink to port 1.0.10		
Set port 1.0.10 for configuring	`Host-A(config)#`	`interface port 1.0.10`
Set the port to access mode	`Host-A(config-if)#`	`switchport mode access`
Associate the port with VLAN10	`Host-A(config-if)#`	`switchport access vlan10`
Add policy map vlink to port 1.0.10	`Host-A(config-if)#`	`service-policy input vlink`
Return to config mode	`Host-A(config-if)#`	`exit`
Step 4. Create an AMF virtual-link tunnel		
Create the virtual link tunnel	`Host-A(config)#`	`atmf virtual-link id 1 ip 192.168.1.1 remote-id 2 remote-ip 192.168.2.1`
Step 5. Create an ACL to permit tunneled traffic		
Enable QoS on switch 1	`Host-A(config)#`	`mls qos enable`
Create an access-list for the virtual link	`Host-A(config)#`	`access-list hardware vlink`
Permit traffic that has the tunneled IP addresses	`Host-A(config-ip-hw-acl)#`	`permit ip 192.168.2.1/32 ip 192.168.1.0/32`
Step 6. Create a class-map for the virtual link		
Create a class-map named vlink	`Host-A(config)#`	`class-map vlink`
Step 7.		
Create a class-map named vlinkarp	`Host-A(config)#`	`class-map vlinkarp`
	`Host-A(config-cmap)#`	`match eth-format ethii-any protocol 0806`
Ensure vlinkarp packets on vlan10 are also sent to the CPU		`match vlan10`
Step 8.		
	`Host-A(config)#`	`policy-map vlink`
		`match access-group`
	`Host-A(config-pmap)#`	`class default`
Send vlink traffic to CoS queue 6		`class vlink remark new-cos 4 both`
Send vlinkarp traffic to CoS queue 6		`class vlinkarp remark new-cos 4 both`

Verifying the AMF Network

To check that all nodes have joined the AMF network use the **show atmf** command on page 86.58 with the summary parameter. You can run this command from any node in an AMF network.

Figure 85-7: Checking AMF configuration using the show atmf summary command

```
AMF_Master#show atmf summary
ATMF Summary Information:
ATMF Status            : Enabled
Network Name           : atmf1
Node Name              : AMF_Master
Role                   : Master
Current ATMF Nodes     : 5
AMF_Master#
```

The **Current AMF Nodes** field in the output above shows that all 5 nodes have joined the AMF network.

Use the **show atmf nodes** command on page 86.82 with the nodes parameter, to check information on individual nodes:

Figure 85-8: Output from the show atmf nodes command

```
AMF_Master#show atmf nodes
Node Information:
  * = Local device
 SC = Switch Configuration:
  C = Chassis   S = Stackable   N = Standalone
Node              Device            ATMF                          Node
Name              Type              Master     SC  Parent        Depth
-------------------------------------------------------------------
* AMF_Master      AT-SBx81CFC400      Y        C   none            0
  Member1         SwitchBlade x908    N        S   AMF_Master      1
  Member2         SwitchBlade x908    N        S   AMF_Master      1
  Member4         x510-52GTX          N        S   Member2         2
  Member3         x510-52GTX          N        S   Member2         2
 Current ATMF node count 5
```

Note that the *Parent* field in the output above refers to the parent *domain* and not the upstream device. In the example output above, Member2 is the domain controller for the parent domain for Member3 and Member4.

Use the **show atmf links** command on page 86.69 to check information on individual AMF links:

Figure 85-9: Checking output with the show atmf links command

```
switch1# show atmf links

ATMF  Links Brief:

Local    Link      Port    ATMF         Adjacent             Adjacent Link
Port     Type      Status  State        Node                 Ifindex  State
-------------------------------------------------------------------------------
sa1      Crosslink Up      TwoWay       Building_1           4501     Forwarding
1.0.1    Downlink  Up      Full         Bld1_Floor_1         5001     Forwarding
1.0.2    Downlink  Up      Full         Bld1_Floor_2         5003     Forwarding
1.0.3    Downlink  Up      Full         Bld2_Floor_1         6101     Forwarding
1.0.4    Crosslink Down    Init         *switch3                      Blocking

* = provisioned
```

Configuring Multiple Nodes at the Same Time: the Unified CLI

The unified CLI is a central component of AMF. It provides you with a configuration and display interface that can control a selected collection of nodes, or the entire AMF network, from a single point. This control is provided through the **atmf working-set** command.

The working-set

An AMF working-set is a set of nodes that can be collectively configured from a single device. Working sets can either **arbitrarily user defined** or **automatically created** (a pre-defined working-set). Specifying or selecting a working-set enables CLI commands to be executed on all nodes within the working-set by using a single command. A working-set can be defined, selected, and configured from any node within an AMF network.

Note

For security reasons you can limit the action of working sets by applying "restrictive login." For more information, see **"atmf restricted-login" on page 86.46**

By default, when you first log into a node that is part of an AMF network, you are implicitly placed into the working-set group **local**, a working-set that contains only the local node. In this instance the CLI prompt when you log in will be either:

- the host-name, if one has been assigned, or
- in the case of a new node in safe mode, a host name based on its MAC address followed by the usual prompt (> or #)

```
Node1> enable
Node1#
```

To create a working set containing a set of nodes use the command **atmf working-set** followed by a comma separated list of the nodes you wish to control. Whenever you select a working set containing any nodes other than the local device, the CLI prompt will display the AMF network name, followed by the number of nodes contained in the working set in square brackets (**atmf1[2]** in the following example).

```
Node1# atmf working-set Node1,Node2
 Node1,Node2
                    Working set join
atmf1[2]#
```

To return to just controlling the local device from any other working set, use the command **atmf working-set group local**.

Working-Set Groups

AMF contains the ability to have working-set groups, so that it is not always necessary to use a comma separated list to specify a working-set.

AMF working-set groups can be split into two types:

- Automatic
- User-defined

Automatic working-set groups

There are three automatic working-set groups that will exist on every AMF network:

1. *All*—all nodes within the AMF network.
2. *Current*—the current working-set of nodes. This group is useful for adding additional nodes to the current working-set.
3. *Local*—the local device

In any AMF network there will also be a number of other automatic working-set groups that are dependent on the platform types which exist within the network. To see the platform dependent automatic working-set groups that exist on the AMF network use the command **show atmf group** with the automatic parameter:

```
x908_VCS_1#show atmf group members automatic

Retrieving Automatic groups from:
x510_1 Master x908_VCS_2 x908_VCS_1

ATMF Group membership

Automatic        Total
Groups           Members   Members
poe              1         Master
x510             1         x510_1
SBx8100          1         Master
x900             2         x908_VCS_2 x908_VCS_1
```

To select a working-set group use the **atmf working-set** command with the group parameter, followed by the group name. You can specify a single group, a comma-separated list of groups, or a comma-separated list of individual nodes followed by a comma-separated list of groups:

```
x908_VCS_1# atmf working-set x510_1,x510_2 group x900

 x510_1, x510_2, x908_VCS_1, x908_VCS_2

                    Working set join

atmf1[4]#
```

If you specify a partially invalid working-set node list or group list, only the valid nodes or groups will join the working set. If you specify a completely invalid working-set, you will create a working-set containing no nodes. The switch will generate a warning message to alert you that the current working-set is empty:

```
atmf1[3]# atmf working-set group x511

% Warning - working set is now empty

atmf1[0]#
```

User-defined working-set groups

In addition to the automatic working-set groups, you can create user-defined groups for arbitrary sets of nodes that you wish to group together, for example, all AMF master nodes.

To create a user-defined working-set group:

1. Create a working-set containing the desired nodes.
2. In global configuration mode use the command **"atmf group (membership)" on page 86.18**.

```
Master# atmf working-set Master1,Master2

Master1,Master2

                    Working set join

atmf1[2]# conf t

atmf1[2]# atmf group new-group-name
```

You can see all user-defined working-set groups that exist on the AMF network with the command **"show atmf group members" on page 86.68**:

```
x908_VCS_1#show atmf group members user-defined

Retrieving Automatic groups from:
x510_1 Master1, Master2, x908_VCS_2 x908_VCS_1

ATMF Group membership

User-defined     Total
Groups           Members   Members
-----------------------------------------------------------------
Masters          2         Master1 Master2

Master#
```

Executing Commands on Working-Sets

Executing commands on a working-set of nodes is very similar to executing commands on a single AlliedWare Plus device.

When a command is executed that is valid for all nodes within the working-set, the output is displayed for each of the nodes separately. However, output will be grouped when it is the same for more than one node.

Here is an example output of the **show arp** command run from a working-set:

```
atmf1[4]#show arp
=======
Master:
=======

 IP Address     MAC Address      Interface    Port       Type
 172.31.0.1     eccd.6d7d.a542   ATMF         sa1        dynamic
 172.31.0.3     0000.cd2b.0329   ATMF         sa1        dynamic
 172.31.0.10    0000.cd37.0163   ATMF         sa1        dynamic

=======
x510_1:
=======

 IP Address     MAC Address      Interface    Port       Type
 172.31.0.2     eccd.6d03.10f9   ATMF         sa4        dynamic

===========
x908_VCS_1:
===========

 IP Address     MAC Address      Interface    Port       Type
 172.31.0.2     0000.cd37.1050   ATMF         sa1        dynamic

===========
x908_VCS_2:
===========

 IP Address     MAC Address      Interface    Port       Type
 172.31.0.2     0000.cd37.1050   ATMF         sa3        dynamic

atmf1[4]#
```

Invalid working-set commands

Some commands can only be executed on *certain* nodes within the working-set. In this case the command will be attempted on all nodes within the working-set. For any node for which the command is not valid, the command execution will fail and the output displayed will indicate the nodes on which the command succeeded and nodes on which the command failed.

Below is example output from the **show card** command run from a working-set, which is only a valid command for the SBx8100 series switches.

```
atmf1[4]# show card
=======
Master:
=======

Slot Card Type        State
-----------------------------------------------
1    AT-SBx81GP24     Online
2    AT-SBx81GP24     Online
3    AT-SBx81GP24     Online
4    AT-SBx81XS6      Online
5    AT-SBx81CFC400   Online (Active)
6    -                -
7    -                -
8    -                -
9    -                -
10   -                -
11   -                -
12   -                -
-----------------------------------------------
==============================
x510_1, x908_VCS_1, x908_VCS_2:
==============================
% Invalid input detected at '^' marker.
```

Sub-configuration limitations for some nodes in a working-set

There will be some instances where a sub-configuration mode is only valid for some of the nodes in the working-set. One example of this would be when entering interface configuration mode for a port that exists on some members of the working-set and not on others. For example:

```
              atmf1[4]# conf t
      atmf1[4](config)# int port1.1.1
% Can't find interface port1.1.1
  atmf1[4:2](config-if)# conf t
```

In the example above the interface **port1.1.1** exists on two of the nodes in the working-set, but does not exist on nodes “Master” or “x510_1”. The interface configuration mode fails for these nodes, and a warning message is output to indicate this. Inside the square brackets, the first number indicates the total number of nodes in the working-set, and the second number indicates the number of nodes in the sub-configuration mode that has been entered. Any configuration commands configured in this mode will only be executed on the nodes that successfully entered the sub-configuration mode.

Entering **exit** while in this mode will return to global configuration mode for all nodes within the working-set:

```
  atmf1[4:2](config-if)# exit
       atmf1[4](config)# (config)#
```

Interactive Commands

It is inappropriate to execute **interactive** commands simultaneously across multiple nodes within a working-set. These commands can only be executed on the local node working-set or on a working-set with a single member.

When any interactive commands are entered from within a working-set they will give an error:

```
atmf1[4]# ping 4.2.2.1
% Working set must contain only single node for this command
```

The list of current interactive commands, including any optional parameters, are:

- ping
- mtrace/mstat
- traceroute
- boot system
- boot configuration-file
- banner login
- tcpdump
- edit
- copy
- mail
- move
- terminal monitor

AMF Backups

AMF backups are a valuable part of AMF network operation. They are the mechanism by which AMF master nodes update their records of the AMF network. By default, AMF master nodes are configured to perform automatic scheduled backups of the entire AMF network once per day at 3.00am. AMF backups can be stored on **remote file servers** or **external removable media** such as USB sticks or SD cards. These backup files can be used in the recovery of a failed node.

Note that this feature will operate only on AMF master nodes.

Using External Media Storage

If storing data on external media, it is a requirement that all AMF masters have external removable media installed with sufficient capacity to hold all of the relevant files stored in the Flash on every node in the AMF network.

Typically a *4 GB* capacity external media storage would be of sufficient size to hold backups for a *40 node* AMF network.

The AMF node backup system has been designed such that the external media used to store the backup data can still be used to store other data. However, care needs to be taken to ensure that enough space is reserved for future AMF backups.

- AMF requires up to 128 MB backup space for SBx8100 nodes and up to 64 MB backup space for other nodes. The output from the **show atmf backup** command on page 86.62 will provide warnings if capacity on the backup media falls below a safe level.

Here is an output example from the **show atmf backup** command showing a backup media space warning:

Figure 85-10: Output showing backup media space warning

```
master1#show atmf backup

Scheduled Backup ...... Disabled
  Schedule ............ 1 per day starting at 12:45
  Next Backup Time .... 25 May 2014 12:45
Backup Media .......... SD (Total 3827.0MB, Free 7.1MB)
                        WARNING: Space on backup media is below 64MB
Current Action ........ Idle
  Started ............. -
  Current Node ........ -
```

Safe removal of external storage media

Removing external storage media, or rebooting the master node, while an AMF backup is underway could potentially cause corruption to files in the backup. Although files damaged as a result of mishandling backup media will be replaced during the next backup cycle, if the file system on the media becomes damaged, it may require reformatting before being inserted into the AMF master. To avoid any damage to the AMF backup files or file system, we recommend that the following procedure be followed before rebooting or removing any external storage media from an AMF master.

1. Disable backups to prevent a scheduled backup from occurring while the card is being removed.

2. Terminate any backup already in process.

3. Verify that it is safe to remove the media by checking that backups are disabled and that there are no backups currently in progress.

Figure 85-11: Example of the safe external storage media removal procedure

```
master1#conf t

master1(config)#no atmf backup enable
master1(config)#exit
master1#atmf backup stop
master1#show atmf backup

Scheduled Backup ...... Disabled
  Schedule ............ 1 per day starting at 12:45
  Next Backup Time .... 25 May 2014 12:45
Backup Media .......... SD (Total 3827.0MB, Free 3257.1MB)
Current Action ........ Idle
  Started ............. -
  Current Node ........ -
```

Once the media has been reinstalled, ensure that the backup scheduler is re-enabled.

Performing a Manual Backup

Whenever a new device is physically added to the AMF network as a provisioned node, we advise that you perform a manual backup from the AMF master.

To perform a manual backup of the entire AMF network, on the AMF master enter the command **atmf backup now** command on page 86.8:

```
Master1# atmf backup now
Master1(config)# atmf backup enable
Master1(config)# exit
```

To check the status of the AMF backup use the **show atmf backup** command on page 86.62.

Figure 85-12: Example output from the show atmf backup command entered during a backup

```
AMF_Master#show atmf backup
Scheduled Backup ...... Enabled
  Schedule ............ 1 per day starting at 03:00
  Next Backup Time .... 14 Dec 2013 03:00
Backup Media .......... USB (Total 3692.6MB, Free 1782.7MB)
Current Action ........ Doing manual backup
  Started ............. 13 Dec 2012 05:20
  Current Node ........ Member1
Node Name                      Date         Time      In ATMF  On Media  Status
-------------------------------------------------------------------------------
AMF_Master                     13 Dec 2012  05:20:16  Yes      Yes       Good
Member1                        -            -         Yes      Yes       -
Member2                        -            -         Yes      No        -
Member3                        -            -         Yes      No        -
Member4                        -            -         Yes      No        -
```

Below is example output from the **show atmf backup** command entered after the backup has completed.

Figure 85-13: Example output from the show atmf backup command entered after backup was completed

```
AMF_Master#show atmf backup
Scheduled Backup ...... Enabled
  Schedule ............ 1 per day starting at 03:00
  Next Backup Time .... 13 Dec 2013 03:00
Backup Media .......... USB (Total 3692.6MB, Free 1651.1MB)
Current Action ........ Idle
  Started ............. -
  Current Node ........ -

Node Name                      Date         Time     In ATMF  On Media  Status
-------------------------------------------------------------------------------
ATMF_Master                 13 Dec 2013  05:20:16  Yes      Yes       Good
Member1                     13 Dec 2013  05:20:27  Yes      Yes       Good
Member2                     13 Dec 2013  05:20:40  Yes      Yes       Good
Member3                     13 Dec 2013  05:20:52  Yes      Yes       Good
Member4                     13 Dec 2013  05:21:08  Yes      Yes       Good
```

Note that the file system used by the AMF backup does not support the backing up of files that have the same name but have different case (e.g. “test.txt” and “TEST.txt”), and only **one** of these files will be stored in the backup. For this reason we recommend that all files on a node be given unique file names.

Backing up to Remote Servers

System backup data can be held on up to two remote backup servers rather than on the master node’s external media. These servers are used for both backup and recovery.

Each AMF master supports a maximum of two remote file servers. The remote backup file servers are mounted on the Master’s file system using SSH and appear as folders.

Configuring a backup to a remote server

First configure the servers. After you have configured the servers you can check the backup media, location, log details and server status using the **show atmf backup** command. You can also manually synchronize the contents of an active server and other configured servers, if required. The following steps show how to set up two backup servers:

1. Use the command **“atmf backup server” on page 86.10** for backup server 1

   This command configures a remote file server(s) as the destination for AMF backups.

   Configuration of a remote server will switch the backup to remote server functionality and disable any further backup to external media. Use the **no** variant of this command to remove the destination servers and revert to backup from external media.

   Note that if no servers are configured, the backup will go to external media. If no servers are configured and no external media exists, no backup will occur.

2. Repeat step (1) for backup server 2
   You should now have two file servers configured to backup your network.

3. Use the **“atmf backup now” on page 86.8** to force a manual backup of your network.

Note This step is optional. Alternatively you could wait until the next scheduled back occurs.

4. Use the command **"show atmf backup" on page 86.62**.

   If you forced a manual backup, you will probably want to display the location and state of each configured file server. The display from this command also shows diagnostic results that test connectivity to each server by using the optional `server-status` parameter.

Below is example output from the **show atmf backup** command showing the configuration of two remote backup file servers.

Figure 85-14: Output from the show atmf backup command showing the configuration of two remote backup file servers

```
x900a#show atmf backup
  Scheduled Backup ...... Enabled
    Schedule ............ 24 per day starting at 14:25
    Next Backup Time .... 19 May 2014 11:25
  Backup Bandwidth ...... Unlimited
  Backup Media .......... FILE SERVER (Total 503837.5MB, Free 186818.0MB)
  Server Config .........
    Synchronization ..... Synchronized
      Last Run .......... 19 May 2014 11:09:50
     1 .................. Configured   (Mounted)
      Host .............. 10.36.150.54
      Username .......... user_1
      Path .............. temp/x900a_1
      Port .............. -
   * 2 .................. Configured   (Mounted, Primary)
      Host .............. tb165.test.com
      Username .......... user_2
      Path .............. temp/x900a_2
      Port .............. -
  Current Action ........ Idle
    Started ............. -
    Current Node ........ -
-------------------------------------------------------------------------------
  Node Name                      Date         Time      In ATMF  On Media  Status
       Synchronization           Date         Time      From Id  To Id     Status
   ----------------------------------------------------------------------------
-
  x210a                          19 May 2014  11:09:37  Yes      Yes       Good
                                 19 May 2014  11:09:46  2        1         Good
  x610a                          19 May 2014  11:09:17  Yes      Yes       Good
                                 19 May 2014  11:09:19  2        1         Good
  x610b                          19 May 2014  11:09:49  Yes      Yes       Good
                                 19 May 2014  11:09:49  2        1         Good
  x610c                          19 May 2014  11:09:20  Yes      Yes       Good
                                 19 May 2014  11:09:20  2        1         Good
  x610d                          19 May 2014  11:09:19  Yes      Yes       Good
                                 19 May 2014  11:09:19  2        1         Good
  x900a                          19 May 2014  11:09:49  Yes      Yes       Good
                                 19 May 2014  11:09:50  2        1         Good
  x908stk                        19 May 2014  11:09:47  Yes      Yes       Good
                                 19 May 2014  11:09:48  Yes      Yes       Good
```

You can use the **show atmf backup** with the parameter `server-status` to display the results of the diagnostics that test connectivity to each server:

Figure 85-15: Output from the show atmf backup command showing diagnostic test results from each server

```
Master1#sh atmf backup server-status
 Id  Last Check  State
  -------------------------------------
 1      186 s  File server ready
  2          1 s  SSH no route to host
```

Node Recovery

Automatic Node Recovery

With AMF, you can replace a failed node with another device and let AMF automatically load the appropriate configuration onto the replacement device.

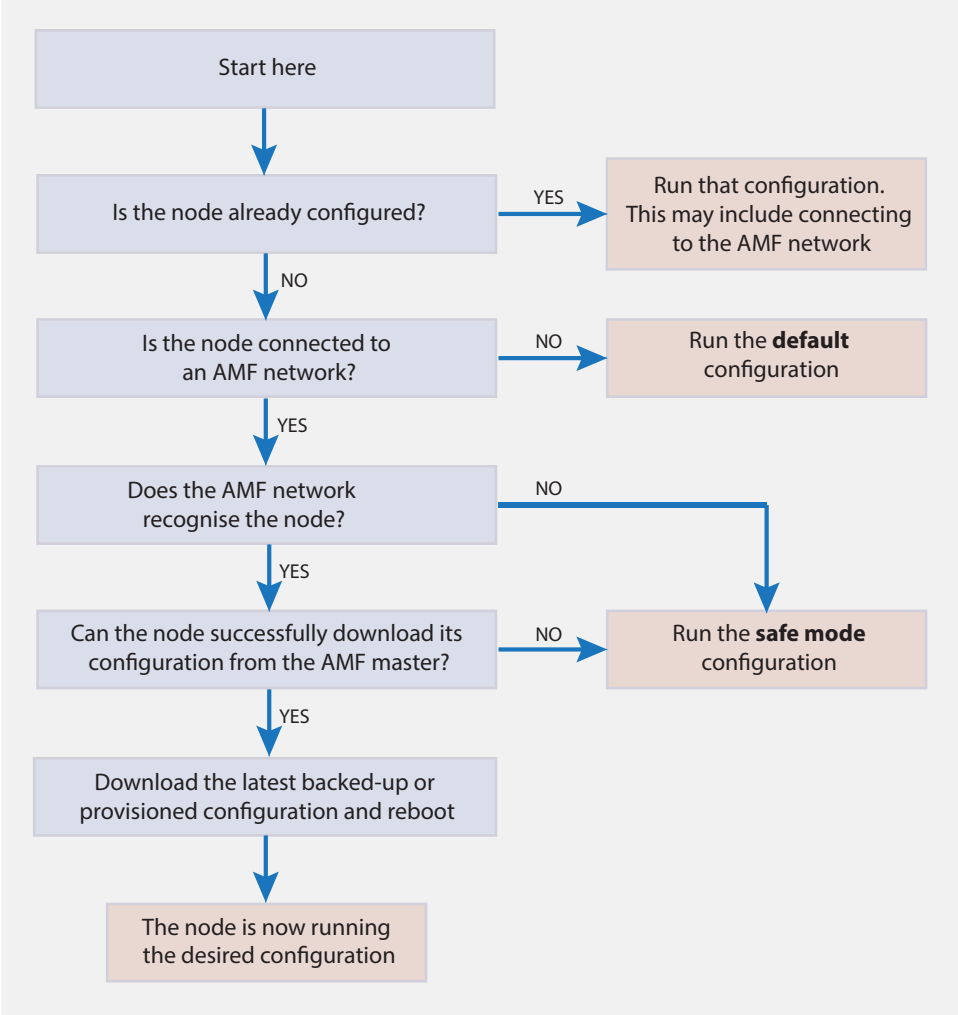
For this to work, the replacement device must have no configuration file. This means it must be either:

- a factory-new device, or
- a used device that has been returned to a "clean" state (see **"Restoring a Node to a "Clean" State" on page 85.36**)

To replace a failed device with a new device of a different platform or with a different node name, you need to provision the network to expect the new device. See **"Node Provisioning" on page 85.4**.

When a switch boots up, it goes through the process in the following flowchart to determine what configuration to use. This flowchart indicates when automatic node recovery will be successful.

Figure 85-16: How a switch determines which configuration to use



Automatic node recovery is not intended to recover multiple nodes simultaneously. If multiple nodes have failed, recover them one at a time.

Caution **Do not make any changes to the device's configuration while a node recovery is underway. A log message will appear on the console or other VTY session indicating when recovery has finished (whether successfully or with errors). This message can also be found by viewing the log with the show log command.**

Figure 85-17: Example log output showing automatic node recovery

```
23:03:15 awplus ATMF[863]: ATMF network detected
23:03:15 awplus ATMF[863]: ATMF safe config applied (forwarding disabled)
23:03:25 awplus ATMF[863]: Shutting down all non ATMF ports
23:03:26 x510_1 ATMF[863]: Automatic node recovery started
23:03:26 x510_1 ATMF[863]: Attempting to recover as x510_1
23:03:26 x510_1 ATMF[863]: Checking master node availability
23:03:32 x510_1 ATMF[863]: Master has joined. 2 members in total.
23:03:32 x510_1 ATMF[863]: x908_VCS_2 has joined. 3 members in total.
23:03:32 x510_1 ATMF[863]: x908_VCS_1 has joined. 4 members in total.
23:03:37 x510_1 ATMFFSR[2950]: Retrieving recovery data from master node Master
23:05:18 x510_1 ATMFFSR[2950]: File recovery from master node succeeded. Node will
now reboot
Flushing file system buffers...
Unmounting any remaining filesystems...
Restarting system.
```

Recovery progress indication

This is a visual feature that displays the recovery status during automatic recovery. This feature uses two distinct flash patterns to indicate the following states:

Recovery State	LED Indication (green)
Recovery in progress	Progressive strobing of all port LEDs.
Recovery failure	All port LEDs alternating on and off, flashing at the same time.

When using this feature during a recovery failure, you can turn off the failure-alert indication and return the port LEDs to their normal running state. To do this, use the command **"atmf recover led-off" on page 86.44**.

You can repeat this process until the recovery failure has been fixed.

Note that the Find me and ecofriendly LED features cannot be used while AMF recovery progress indication is active.

Restoring a Node to a "Clean" State

When replacing a failed device, your replacement device should be one of the following types, in order for AMF automatic node recovery to work:

- A factory-new device
- A used device that has been returned to a "clean" state

A clean device is one that has had its previous configuration components removed. The process of cleaning is required when replacing a failed device with one that, although in working condition, has been used previously and still retains components of its previous configuration.

If you keep on-site spares, store them with clean configurations and current releases. When you upgrade your network to a new AlliedWare Plus version, we recommend you upgrade your spare devices too.

To clean up a previously used device, use the **atmf cleanup** command on page 86.13. This command erases all data from NVS and Flash **apart from**:

- The boot release file (a .rel file) and its release setting file
- v1 license files /.configs/.swfeature.lic
- v2 license files /.configs/.sw_v2.lic

The device is then rebooted to put it into a clean state. The device can then be used for automatic node recovery.

Any other user files that remain in Flash will be overwritten during the automatic recovery process. If there are any files stored in the Flash of the replacement device that need to be retained, back these files up prior to installing the device into the AMF network.

Manual Node Recovery

There are certain situations where automatic recovery may fail. Automatic recovery has been designed to be cautious in its approach to recovering nodes for reasons such as:

- The backup stored on the AMF master not having a "Good" status
- The replacement device having a release of the AlliedWare Plus Operating System installed on it that is old enough to be incompatible with AW+ on the neighbor or the master.

When these situations occur, automatic node recovery will fail.

In this failed state, the replacement device will have the AMF safe configuration mode applied (see **"AMF Safe Configuration Procedures" on page 85.39**). After investigating the failure and taking remedial action, you may want to initiate manual node recovery. To do this, enter the following command:

```
amf1# atmf recover {<node_name>} {<master_node_name>}
```

where:

- **node_name** is the host name of the device you wish to recover.
- **master_node_name** is the host name of the AMF master that contains the backup you want to use for the recovery.

The manual recovery command will bypass the usual checks performed by automatic node recovery. Make sure that the backup configuration stored on the specified AMF master is correct before you execute the command.

If you attempt to manually recover a node with the backup file of a node from a **different platform**, the release file from the backup will be incompatible and won't be copied to the replacement device. Instead, the existing release on the replacement device will be used, in order to ensure the device can join the AMF network and function correctly.

Figure 85-18: Example output showing manual recovery

```
amf1#atmf recover x510_1 Master
This command will erase ALL flash contents.  Continue node recovery? (y/n)y
Manual node recovery successfully initiated
x510_1#23:15:32 x510_1 ATMFFSR[8477]: Retrieving recovery data from master node
Master
23:17:17 x510_1 ATMFFSR[8477]: Manual node recovery completed
x510_1#
```

Node Recovery on VCStacks

Node recovery on VCStacks that are part of an AMF network is somewhat different to node recovery of standalone devices.

This is because VCStack has its own node recovery mechanism that has different requirements to AMF.

In the extremely unlikely situation of needing to replace an entire VCStack that is a member of an AMF network, you can use AMF automatic node recovery to first recover Stack ID 1, which will become the VCStack master.

The replacement device which will become the VCStack master must be a clean unit (see **"Restoring a Node to a "Clean" State" on page 85.36**).

The procedure for recovering an entire stack is as follows:

1. Connect a clean device to the AMF network, and power it on. The connections into the AMF network should be between the appropriately configured AMF links on the neighboring node, and the ports previously configured as AMF links in the backup for the failed node configuration.
2. The AMF network should detect the replacement device and begin automatic node recovery. Wait until automatic node recovery completes, then check that the replacement device has come up correctly as VCStack ID 1, and that the configuration is correct.
3. Configure the next replacement device as VCStack ID 2. Ensure it is installed with a compatible release and the same set of licenses that exist on ID 1. Connect the VCStack cables and power it on.
4. VCStack ID 1 should detect ID 2 and synchronize the configuration and firmware release. Once this has completed, check that the VCStack has formed correctly, and then connect the remaining network connections.

For any additional VCStack members, repeat the last two steps, ensuring that the VCStack ID is set to the next sequential value for each additional device that is added to the VCStack.

For more information on VC Stacking, see:

- **"VCStack Introduction" on page 110.2**
- **"Replacing a stack member" on page 110.15**

AMF Safe Configuration

If AMF automatic node recovery fails, AMF contains a safety net feature that puts the replacement node into a safe configuration state. This is to prevent an unconfigured device from joining the network and creating loops.

Detecting AMF Safe Configuration Operation

A log message will be generated when AMF safe configuration is applied. This message will appear in the log some time after the startup sequence.

The message will also be output to the console or any connected VTY session.

AMF Safe Configuration Procedures

The procedures for AMF safe configuration are shown below:

- A special VLAN is created in the disabled state and given the name atmf_node_recovery_safe_vlan. The VID of this VLAN is determined dynamically to ensure that it does not conflict with either of the AMF management VLANs, or any other VLANs that are detected on the AMF network.
- All ports are removed from their default VLAN membership (VLAN 1).
- All ports are set as tagged members of the safe VLAN.
- Additionally, all ports that are not an AMF link or cross-link are shut down. The links and crosslinks are detected by AMF and added to the dynamic configuration. This is done to ensure correct behavior of static aggregators and Layer 3 protocols configured on the neighboring devices.

Figure 85-19: Example output from the show vlan command with the brief parameter set for a device in AMF safe configuration mode

```
awplus#sh vlan brief

VLAN ID  Name            Type    State   Member ports   (u)-Untagged, (t)-Tagged
======= ================ ======= ======= =======================================
1       default          STATIC  ACTIVE
4090    atmf_node_recovery_safe_vlan
                         STATIC  SUSPEND port1.0.1(t)  port1.0.2(t)  port1.0.3(t)
                                         port1.0.4(t)  port1.0.5(t)  port1.0.6(t)
                                         port1.0.7(t)  port1.0.8(t)  port1.0.9(t)
                                         port1.0.10(t) port1.0.11(t)
                                         port1.0.12(t) port1.0.13(t)
                                         port1.0.14(t) port1.0.15(t)
                                         port1.0.16(t) port1.0.17(t)
                                         port1.0.18(t) port1.0.19(t)
                                         port1.0.20(t) port1.0.21(t)
                                         port1.0.22(t) port1.0.23(t)
                                         port1.0.24(t)
```

Figure 85-20: Example output from the show running-config command for a device in AMF safe configuration mode

```
awplus#show running-config
...
!
vlan database
 vlan 4090 name atmf_node_recovery_safe_vlan
 vlan 4090 state disable
!
interface port1.0.1-1.0.4
 shutdown
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 4090
 switchport trunk native vlan none
!
interface port1.0.5
 switchport
 switchport atmf-link
 switchport mode trunk
 switchport trunk allowed vlan add 4090
 switchport trunk native vlan none
!
interface port1.0.6-1.0.24
 shutdown
 switchport
 switchport mode trunk
 switchport trunk allowed vlan add 4090
 switchport trunk native vlan none
!
...
```

Undoing an AMF Safe Configuration

If your node has had AMF safe configuration applied, you can use normal CLI configuration commands to modify the running-configuration to whatever configuration is required.

See below for an example of returning a device from AMF safe configuration mode to having default VLAN and port settings. Note that in this example a 24-port device has been used.

```
awplus# configure terminal
awplus(config)# interface port1.0.1-port1.0.24
awplus(config-if)# switchport trunk native vlan 1
awplus(config-if)# switchport trunk allowed vlan remove 4090
awplus(config-if)# switchport mode access
                % port1.0.5 has ATMF link configured so
                its mode cannot be changed
```

```
awplus(config-if)# no shutdown
awplus(config-if)# exit
awplus(config-if)# vlan database
awplus(config-if)# no vlan 4090
awplus(config-if)# end
```

In order to retain connectivity to the AMF network, AMF link and crosslink settings should not be changed. In the example above you can see that port1.0.5 is an automatically configured AMF link. You can also see the error message indicating it was skipped by the **"switchport mode access" on page 19.18**. This is because AMF links must be in trunk mode.

Caution

No changes should be made to the device's configuration while a node recovery is underway. A log message will appear on the console or other logged in session indicating when recovery has finished (whether successfully or with errors). This message can also be found by viewing the log with the command "show log" on page 12.40.

Rolling-Reboot Firmware Upgrade

The rolling-reboot firmware upgrade feature enables nodes within an AMF network to be rebooted and upgraded in a rolling sequence so that downtime and management overheads are minimized. First, specify a set of nodes within the AMF network using the **atmf working-set** command, then use the **atmf reboot-rolling** command. All nodes in the specified working-set will be rebooted and upgraded one by one, starting with the nodes furthest from the core domain, and ending with nodes closest to, or in, the core domain.

Once the rebooted node has finished running its configuration and has brought its ports up, it re-joins the AMF network and the next node in the working-set is rebooted and upgraded.

Note The **atmf rolling-reboot** command can also be used to reboot a set of nodes without upgrading the firmware.

To upgrade firmware, a download URL can be selected from any media location.

Supported media locations include:

- flash:
- card:
- usb:
- tftp:
- scp:
- http:

The latest compatible release for a node will be selected from one of these locations. Several checks need to be performed to ensure the upgrade will succeed. This includes checking that the current node release boots from Flash and that there is enough space in Flash on this node. The new release name is updated using the **boot system backup** command. The old release will become the backup release file.

Note that if the release file is to be copied from a remote location (e.g. via TFTP or HTTP), the URL should specify the exact release filename without using wild card characters.

The node is rebooted and the new software version will be used. On bootup, the software release is verified. Should an upgrade fail, the upgrading unit will fail back to its old software version. At the completion of this command, a report is run showing the release upgrade status of each node.

The **force** parameter enforces a node reboot, even though the node may not be suitable for upgrading software. This command can take a significant amount of time to complete.

Note Rolling-reboot firmware upgrades can be performed on a working-set that includes the controlling node, although in this instance the user will not be presented with a summary report upon completion.

Here is an example of a Rolling-reboot firmware upgrade summary report:

```
================================================================================
ATMF Rolling Reboot Complete
Node Name   Reboot Status    Release Name                Release Status
--------------------------------------------------------------------------------

Node1       Rebooted         x510-main-20121018-2.rel    Upgraded

Node2       Rebooted         x900-main-20121018-2.rel    Upgraded

Node3       Rebooted         x900-main-20121018-2.rel    Upgraded

Node4       Rebooted         x510-main-20121018-2.rel    Upgraded
================================================================================
```

Performing a Rolling-Reboot Upgrade

To perform a Rolling-reboot firmware upgrade on all nodes in the AMF network, first select all nodes using the default working-set group **all**:

```
SBSBx8100# atmf working-set group all

SBx8100, SBx908-VCS1, SBx908-VCS2, x510_1, x510_2:

                    Working set join
```

Next, using the **atmf reboot-rolling** command, specify the path to the release files to use for the upgrade. In the following example the release files are stored on the external USB storage media installed in the node controlling the Rolling-reboot firmware upgrade, in a directory called "rel". Note that because the node controlling the Rolling-reboot firmware upgrade is included in the nodes to be upgraded, a message is output indicating that no summary will be available on completion.

```
csg_vcf[5]#atmf reboot-rolling usb:/rel/*.rel
Retrieving data from SBSBx8100
Retrieving data from SBx908-VCS2
Retrieving data from x510_1
Retrieving data from x510_2
Retrieving data from SBx908-VCS1

ATMF Rolling Reboot Nodes:

                    Timeout
Node Name           (Minutes)  New Release File           Status
-----------------------------------------------------------------------
x510_2              9          x510-main-20121203-1.rel   Release ready
x510_1              6          x510-main-20121203-1.rel   Release ready
SBx908-VCS1         9          x900-main-20121203-1.rel   Release ready
SBx908-VCS2         9          x900-main-20121203-1.rel   Release ready
SBSBx8100           11         SBx81CFC400-main-20121203  Release ready
                                 -1.rel
% The controlling node (SBSBx8100) is included in the
rolling reboot and will be rebooted last.
No summary will be available on completion.
Continue upgrading releases ? (y/n):
=================================================================
Copying Release    : x510-main-20121203-1.rel to x510_2
Updating Release   : x510-main-20121203-1.rel information on x510_2
=================================================================
ATMF Rolling Reboot: Rebooting x510_2
=================================================================
02:11:32 SBSBx8100 ATMF[1973]: x510_2 has left. 4 members in total.

% x510_2 has left the working-set
02:13:30 SBSBx8100 ATMF[1973]: x510_2 has joined. 5 members in total.
Reboot of x510_2 has completed
```

Although in this example no summary report was generated, you can refer to the progress messages output on the console to confirm that the upgrades were successful. You can also use the **atmf working-set** and the **show boot** commands to confirm the current boot image for each node in the AMF network.

```
================================================================
Copying Release    : x510-main-20121203-1.rel to x510_1
Updating Release   : x510-main-20121203-1.rel information on x510_1
================================================================
ATMF Rolling Reboot: Rebooting x510_1
================================================================
02:14:13 SBSBx8100 ATMF[1973]: x510_1 has left. 4 members in total.

% x510_1 has left the working-set
02:15:53 SBSBx8100 ATMF[1973]: x510_1 has joined. 5 members in total.
Reboot of x510_1 has completed

================================================

Copying Release    : x900-main-20121203-1.rel to SBx908-VCS1
Updating Release   : x900-main-20121203-1.rel information on SBx908-VCS1
================================================================
ATMF Rolling Reboot: Rebooting SBx908-VCS1
================================================================
02:19:02 SBSBx8100 ATMF[1973]: x510_1 has left. 4 members in total.
02:19:02 SBSBx8100 ATMF[1973]: SBx908-VCS1 has left. 3 members in total.

% SBx908-VCS1 has left the working-set
02:20:48 SBSBx8100 ATMF[1973]: SBx908-VCS1 has joined. 4 members in total.
Reboot of SBx908-VCS1 has completed
02:20:51 SBSBx8100 ATMF[1973]: x510_1 has joined. 5 members in total.
================================================================
Copying Release    : x900-main-20121203-1.rel to SBx908-VCS2
Updating Release   : x900-main-20121203-1.rel information on SBx908-VCS2
================================================================
ATMF Rolling Reboot: Rebooting SBx908-VCS2
================================================================
02:21:54 SBSBx8100 ATMF[1973]: x510_2 has left. 4 members in total.
02:21:54 SBSBx8100 ATMF[1973]: SBx908-VCS2 has left. 3 members in total.

% SBx908-VCS2 has left the working-set
02:23:35 SBSBx8100 ATMF[1973]: SBx908-VCS2 has joined. 4 members in total.
Reboot of SBx908-VCS2 has completed
================================================================
Copying Release    : SBx81CFC400-main-20121203-1.rel to SBSBx8100
02:23:39 SBSBx8100 ATMF[1973]: x510_2 has joined. 5 members in total.
Updating Release   : SBx81CFC400-main-20121203-1.rel information on SBSBx8100
================================================================
ATMF Rolling Reboot: Rebooting SBSBx8100
================================================================
02:24:07 SBSBx8100 ATMF: reboot-rolling Rebooting SBSBx8100 at request of user
manager.
```

Chapter 86: AMF Commands

Introduction 86.3
AMF Naming Convention 86.3
atmf backup 86.4
atmf backup bandwidth 86.5
atmf backup delete 86.6
atmf backup enable 86.7
atmf backup now 86.8
atmf backup server 86.10
atmf backup stop 86.12
atmf backup synchronize 86.12
atmf cleanup 86.13
atmf distribute firmware 86.14
atmf domain vlan 86.16
atmf enable 86.17
atmf group (membership) 86.18
atmf log-verbose 86.20
atmf management subnet 86.21
atmf management vlan 86.23
atmf master 86.25
atmf network-name 86.26
atmf provision 86.27
atmf provision node clone 86.28
atmf provision node configure boot config 86.30
atmf provision node configure boot system 86.31
atmf provision node create 86.32
atmf provision node delete 86.34
atmf provision node license-cert 86.36
atmf provision node locate 86.38
atmf reboot-rolling 86.39
atmf recover 86.43
atmf recover led-off 86.44
atmf remote-login 86.45
atmf restricted-login 86.46
atmf virtual-link id ip remote-id remote-ip 86.47
atmf working-set 86.49
clear atmf links statistics 86.51
debug atmf 86.52
debug atmf packet 86.54
erase factory-default 86.57
show atmf 86.58
show atmf backup 86.62
show atmf detail 86.64
show atmf group 86.66
show atmf group members 86.68
show atmf links 86.69
show atmf links detail 86.70
show atmf links statistics 86.77

show atmf memory 86.80
show atmf nodes 86.82
show atmf provision nodes 86.83
show atmf tech 86.84
show atmf working-set 86.86
show debugging atmf 86.86
show debugging atmf packet 86.87
show running-config atmf 86.87
switchport atmf-crosslink 86.88
switchport atmf-link 86.90
type atmf node 86.91

Introduction

This chapter provides an alphabetical reference for AMF commands.

AMF Naming Convention

When AMF is enabled on a switch, it will automatically be assigned a host name. If a host name has already been assigned, by using the command **"hostname" on page 10.18**, this will remain. If however, no host name has been assigned, then the name applied will be the prefix, **host**_ followed (without a space) by the MAC address of the device. For example, a device whose MAC address is **0016.76b1.7a5e** will have the name **host_0016_76b1_7a5e** assigned to it.

To efficiently manage your network using AMF, we strongly advise that you devise a naming convention for your network switches, and accordingly apply an appropriate hostname to each switch in your AMF network.

atmf backup

This command can only be applied to a master node. It manually schedules an AMF backup to start at a specified time and to execute a specified number of times per day.

Use the **no** variant of this command to disable the schedule.

Syntax

```
atmf backup {default|<hh:mm> frequency <1-24>}
no atmf backup enable
```

Parameter	Description
`default`	Restore the default backup schedule.
<hh:mm>	Sets the time of day to apply the first backup, in hours and minutes. Note that this parameter uses the 24 hour clock.
`backup`	Enables AMF backup to external media.
`frequency <1-24>`	Sets the number of times within a 24 hour period that backups will be taken.

Default Backups run daily at 03:00 AM, by default

Mode Global Configuration

Usage Running this command only configures the schedule. To enable the schedule, you should then apply the command **atmf backup enable**.

Example To schedule backup requests to begin at 11 am and execute twice per day (11 am and 11 pm), use the following command:

```
VCF_1# configure terminal
VCF_1(config)# atmf backup 11:00 frequency 2
```

Caution File names that comprise identical text, but with differing case, such as Test.txt and test.txt, will not be recognized as being different on a FAT32 based backup media such as a USB storage device. However, these filenames will be recognized as being different on your Linux based switch. Therefore, for good practice, ensure that you apply a consistent case structure for your back-up file names.

Related Commands

atmf backup enable
atmf backup stop
show atmf backup

atmf backup bandwidth

This command sets the maximum bandwidth in kilobytes per second (kBps) available to the AMF backup process. This command enables you to restrict the bandwidth that is utilized for downloading file contents during a backup.

Note

This command will only run on an AMF master. An error message will be generated if the command is attempted on node that is not a master.

Also note that setting the bandwidth value to zero will allow the transmission of as much bandwidth as is available, which can exceed the maximum configurable speed of 1000 kBps. In effect, zero means unlimited.

Use the **no** variant of this command to reset (to its default value of zero) the maximum bandwidth in kilobytes per second (kBps) available when initiating an AMF backup. A value of zero tells the backup process to transfer files using unlimited bandwidth.

Syntax

```
atmf backup bandwidth <0-1000>

no atmf backup bandwidth
```

Parameter	Description
<0-1000>	Sets the bandwidth in kilobytes per second (kBps)

Default The default value is zero, allowing unlimited bandwidth when executing an AMF backup.

Mode Global Configuration

Examples To set an atmf backup bandwidth of 750 kBps, use the commands:

```
node2# configure terminal
node2(config)# atmf backup bandwidth 750
```

To set the atmf backup bandwidth to the default value for unlimited bandwidth, use the commands:

```
node2# configure terminal
node2(config)# no atmf backup bandwidth
```

Related Commands **show atmf backup**

atmf backup delete

This command removes the backup file from the external media of a specified AMF node.

Syntax `atmf backup delete <node name>`

Parameter	Description
<node name>	The AMF node name of the backup file to be deleted.

Mode Privileged Exec

Example To delete the backup file from node2, use the following command:

```
Node_1# atmf backup delete node2
```

Related Commands
show atmf backup
atmf backup now
atmf backup stop

atmf backup enable

This command enables automatic AMF backups on the AMF master node that you are connected to. By default, automatic backup starts at 3:00 AM. However, this schedule can be changed by the **atmf backup** command on page 86.4. Note that backups are initiated and stored only on the master nodes.

Use the **no** variant of this command to disable any AMF backups that have been scheduled and previously enabled.

Syntax

```
atmf backup enable
no atmf backup enable
```

Default Automatic AMF backup functionality is enabled on the AMF master when it is configured and external media, i.e. an SD card or a USB storage device or remote server, is detected.

Mode Global Configuration

Usage A warning message will appear if you run the **atmf backup enable** command with either insufficient or marginal memory availability on your external storage device.

You can use the command **"show atmf backup" on page 86.62** to check the amount of space available on your external storage device.

Example To turn on automatic AMF backup, use the following command:

```
AMF_Master_1# configure terminal
AMF_Master_1(config)# atmf backup enable
```

Related Commands

show atmf
show atmf backup
atmf backup
atmf backup now
atmf enable

atmf backup now

This command initiates an immediate AMF backup of either all AMF members, or a selected AMF member. Note that this backup information is stored in the external media on the master node of the switch on which this command is run, even though the selected AMF member may not be a master node.

Syntax `atmf backup now [<nodename>]`

Parameter	Description
<nodename> or *<hostname>*	The name of the AMF member to be backed up - as set by the command **"hostname" on page 10.18**. Where no name has been assigned to this device, then you must apply the prefix, host underscore followed (without a space) by the MAC address of the device to be backed up. For example `host_0016_76b1_7a5e` Note that the node-name appears as the command Prompt when in Privileged Exec mode.

Default A backup is initiated for all nodes on the AMF (but stored on the master nodes).

Mode Privileged Exec

Usage Although this command will select the AMF node to be backed-up; it can only be run from any AMF master node.

Note The backup produced will be for the selected node but the backed-up config will reside on the external media of the AMF master node on which the command was run. However, this process will result in the information on one master being more up-to-date. To maintain concurrent backups on both masters, you can apply the backup now command to the master working-set. This is shown in **"Example 4" on page 86.9**.

Example 1 In this example, an AMF member has not been assigned a host name. The following command is run on the AMF_Master_2 node to immediately backup the device - identified by its MAC address of 0016.76b1.7a5e:

```
AMF_Master_2# atmf backup now host_0016_76b1_7a5e
```

Note When a host name is derived from its MAC address, the syntax format entered changes from XXXX.XXXX.XXXX to XXXX_XXXX_XXXX.

Example 2 In this example, an AMF member has the host name, **office_annex**. The following command will immediately backup this device:

```
AMF_Master_2# atmf backup now office_annex
```

This command is initiated on the switch's master node named **AMF_Master_2** and initiates an immediate backup on the switch named **office_annex**.

Example 3 To initiate from AMF_master_1 an immediate backup of all AMF member nodes, use the following command:

```
AMF_Master_1# amf backup now
```

Example 4 To initiate an immediate backup of the node with the host-name “office_annex” and store the configuration on both masters, use the following process:

From the AMF_master_1, set the working-set to comprise only of the automatic group, master nodes.

```
AMF_Master_1# atmf working-set group master
```

This command returns the following display:

```
============================
AMF_Master_1, AMF_Master_2
============================

Working set join
```

Backup the AMF member with the host name, **office_annex** on both the master nodes as defined by the working set.

```
AMF_Master[2]# atmf backup now office_annex
```

Note that the [2] shown in the command prompt indicates a 2 node working-set.

Related Commands

atmf backup
atmf backup stop
hostname
show atmf backup

atmf backup server

This command configures remote file servers as the destination for AMF backups.

Use the **no** variant of this command to remove the destination server(s). When all servers are removed the system will revert to backup from external media.

Syntax

```
atmf backup server id {1|2} <hostlocation> username <username>[path
    <path>|port <1-65535>]

no atmf backup server id {1|2}
```

Parameter	Description
id	Remote server backup server identifier.
{1|2}	The backup server identifier number (1 or 2). Note that there can be up to two backup servers, numbered 1 and 2 respectively, and you would need to run this command separately for each server.
<hostlocation>	Either the name or the IP address (IPv4 or IPv6) of the selected backup server (1 or 2).
username	Configure the username to log in with on the selected remote file server.
<username>	The selected remote file server's username.
path	The location of the backup files on the selected remote file server. By default this will be the home directory of the username used to log in with.
<path>	The directory path utilized to store the backup files on the selected remote file server. No spaces are allowed in the path.
port	The connection to the selected remote backup file server using SSH. By default SSH connects to a device on TCP port 22 but this can be changed with this command.
<1-65535>	A TCP port within the specified range.

Defaults Remote backup servers are not configured. The default SSH TCP port is 22. The path utilized on the remote file server is the home directory of the username.

Mode Global Exec

Usage The hostname and username parameters must both be configured.

Examples To configure a remote backup server at 192.168.1.1 with the login username of backup1, the backup repository on atmf/network/location/ port 1024, use the command:

```
AMF_Master_1# configure terminal
AMF_Master_1(config)# atmf backup server id 1 192.168.1.1
                      username backup1 path atmf/network/
                      location/port 1024
```

To configure server 1 with an ipv4 address and a username of backup1, use the commands:

```
AMF_Master_1# configure terminal
AMF_Master_1(config)# atmf backup server id 1 192.168.1.1
username backup1
```

To configure server 1 with an ipv6 address and a username of backup1, use the command:

```
AMF_backup1_1# configure terminal
AMF_Master_1(config)# atmf backup server id 1 FFEE::01 username
backup1
```

To configure server 2 with a hostname and username, use the command:

```
AMF_Master_1# configure terminal
AMF_Master_1(config)# atmf backup server id 2 www.example.com
username backup2
```

To configure server 2 with a hostname and username in addition, the optional path and port parameters, use the command:

```
AMF_Master_1# configure terminal
AMF_Master_1(config)# atmf backup server id 2 www.example.com
username backup2 path tokyo port 1024
```

To unconfigure the AMF remote backup file server 1, use the command:

```
AMF_Master_1# configure terminal
AMF_Master_1(config)# no atmf backup server id 1
```

Related Commands **show atmf backup**

atmf backup stop

Running this command stops a backup that is currently running on the master node you are logged onto. Note that if you have two masters and want to stop both, then you can either run this command separately on each master node, or add both masters to a working set, and issue this command to the working set.

Syntax `atmf backup stop`

Mode Privileged Exec

Usage This command is used to halt an AMF backup that is in progress. In this situation the backup process will finish on its current node and then stop.

Example To stop a backup that is currently executing on master node VCF-1, use the following command:

```
AMF_Master_1# amf backup stop
```

Related Commands
atmf backup
atmf backup enable
atmf backup now
show atmf backup

atmf backup synchronize

For the master node you are connected to, this command initiates a system backup of files from the node's active remote file server to its backup remote file server. Note that this process happens automatically each time the network is backed up.

Syntax `atmf backup synchronize`

Mode Privileged Exec

Example When connected to the master node AMF_Master_1, the following command will initiate a backup of all system related files from its active remote file server to its backup remote file server.

```
AMF_Master_1# atmf backup synchronize
```

Related Commands
show atmf backup
atmf backup enable
show atmf

atmf cleanup

This command erases all data from NVS and all data from Flash **excluding** the following:

- The current release file and its /flash/.release file
- The backup release file and /flash/.backup file
- v1 license files /flash/.configs/.swfeature.lic
- v2 license files /flash/.configs/.sw_v2.lic

It then reboots to put the device in a clean state ready to be used as a replacement node on a provisioned port.

Syntax `atmf cleanup`

Mode Privileged Exec

Usage This command is an alias to the **erase factory-default** command on page 86.57.

Example To erase data, use the command:

```
Node_1(config)# atmf cleanup

This command will erase all NVS, all flash contents except
for the boot release, and any license files, and then
reboot the switch. Continue? (y/n):y
```

Related Commands **erase factory-default**

atmf distribute firmware

This command can be used to upgrade software one AMF node at a time. A URL can be selected from any media location. The latest compatible release for a node will be selected from this location.

Several procedures are performed to ensure the upgrade will succeed. This includes checking the current node release boots from flash. If there is enough space on flash the software release is copied to flash on the new location.

The new release name is updated using the **boot system** command. The old release will become the backup release file. If a release file exists in a remote device (such as TFTP or HTTP, for example) then the URL should specify the exact release filename without using a wild card character.

The command will continue to upgrade software until all nodes are upgraded. At the end of the upgrade cycle the **reboot** command should be used on the working-set.

Syntax `atmf distribute firmware <url>`

Parameter	Description
<url>	The URL of the file. See **"URL Syntax" on page 6.12** for valid URL syntax.

Mode Privileged Exec

Examples To upgrade nodes in a atmf network with a predefined AMF group called `sw_team`, use the following commands:

```
SW_Team1# atmf working-set group sw_team
```

Output

```
============================
SW_Team1, SW_Team2, SW_Team3:
============================

Working set join
```

```
ATMF_NETWORK[3]# atmf distribute firmware card:*.rel
```

Output

```
Retrieving data from SW_Team1
Retrieving data from SW_Team2
Retrieving data from SW_Team3

ATMF Firmware Upgrade:


Node Name          New Release File           Status
-----------------------------------------------------------------
SW_Team1           x510-main-20140204-2.rel   Release ready
SW_Team2           x610-main-20140204-2.rel   Release ready
SW_Team3           x610-main-20140204-2.rel   Release ready
Continue the rolling reboot ? (y/n):y
=================================================================
Copying Release    : x510-main-20140204-2.rel to SW_Team1
Updating Release   : x510-main-20140204-2.rel information on SW_Team1
=================================================================
Copying Release    : x610-main-20140204-2.rel to SW_Team2
Updating Release   : x610-main-20140204-2.rel information on SW_Team2
=================================================================
Copying Release    : x610-main-20140204-2.rel to SW_Team3
Updating Release   : x610-main-20140204-2.rel information on SW_Team3

=================================================================
New firmware will not take effect until nodes are rebooted.
=================================================================

ATMF_NETWORK[3]#
```

Related Commands atmf working-set

atmf domain vlan

The AMF domain vlan is one of the internal VLANs that are used to communicate information about the state of the AMF network between nodes. AMF uses its internal VLANS (the management VLAN and the domain VLAN) to communicate its inter nodal network status information. These VLANs must be reserved for AMF and not used for other purposes.

When an AMF network is first created all its nodes are assigned a domain VLAN with a default (domain) VID of 4091. An important point conceptually is that although this VLAN then exists globally across the AMF network, it is assigned separately to each domain. The AMF network therefore can be thought of as comprising a series of domain VLANS each having the same VID and each being applied to a horizontal slice (domain) of the AMF. It follows therefore that the domain VLANs are only applied to ports that form cross-links and not to ports that form uplinks/downlinks.

If you assign a VLAN ID to this VLAN (i.e. changing its value from the default of 4091) then you will need to do this separately on every device within the AMF network. The AMF domain subnet will then be applied to this new VID when all devices within the AMF network are next rebooted.

Use the **no** variant of this command to reset the VLAN ID to its default value of 4091.

Syntax

```
atmf domain vlan <2-4090>
no atmf domain vlan .
```

Parameter	Description
<2-4090>	The VLAN number in the range 2 to 4090.

Default The default domain VLAN ID for the AMF is 4091.

Mode Global Configuration

Usage The VLANs involved in this process, must be reserved for AMF and cannot be used for other purposes. This command enables you to change the domain VLAN to match your network's specific configuration.

Caution

Setting this command, then rebooting the switch will only apply the AMF VLAN for the switch being configured. The new domain vlan will not become effective for the AMF network until all its member nodes have been updated, and all its member switches rebooted.

As part of its automatic creation process, this VLAN will also be assigned an IP subnet address based on the value configured by the command **"atmf management subnet" on page 86.21**. Refer to this command for more information.

Examples To change the AMF domain VLAN to 4000 use the following commands:

```
VCF-1# configure terminal
VCF-1(config)# atmf domain vlan 4000
```

To reset the AMF domain VLAN to its default of 4091, use the following commands:

```
VCF-1# configure terminal
VCF-1(config)# no atmf domain vlan
```

atmf enable

This command manually enables (turns on) the AMF feature for the switch being configured.

Use the **no** variant of this command to disable (turn off) the AMF feature on the member node.

Syntax

```
atmf enable
no atmf enable
```

Default Once AMF is configured, the AMF feature starts automatically when the switch starts up.

Mode Global Configuration

Usage The switch does not auto negotiate AMF domain specific settings such as the Network Name. You should therefore, configure your switch with any domain specific (non default) settings before enabling AMF.

Examples To turn on the AMF the feature:

```
MyNode# config terminal
MyNode(config)# atmf enable
```

To turn off the AMF feature:

```
MyNode(config)# no atmf enable
```

This command returns the following display:

```
% Warning: The ATMF network config has been set to enable
% Save the config and restart the system for this change to take
effect.
```

atmf group (membership)

This command configures a switch to be a member of one or more AMF groups. Groups exist in three forms: Implicit Groups, Automatic Groups, and User-defined Groups.

- Implicit Groups
  - « all - All nodes in the AMF
  - « current - The current working-set
  - « local - The originating node.

  Note that the Implicit Groups do not appear in show group output.
- Automatic Groups - These are defined by hardware architecture, e.g. x510, x610, x900, x8100.
- User-defined Groups - These enable you to define arbitrary groups of AMF members based on your own criteria.

Each node in the AMF is automatically assigned membership to the implicit groups, and the automatic groups that are appropriate to its node type, e.g. x610, PoE. Similarly, nodes that are configured as masters are automatically assigned to the master group.

Use the **no** variant of this command to remove the membership.

Syntax

```
atmf group <group-list>
no atmf group <group-list>
```

Parameter	Description
<group-list>	A list of group names. These should be entered as a comma delimited list without spaces.

Mode Global Configuration

Usage You can use this command to define your own arbitrary groups of AMF members based on your own network's configuration requirements. Applying a node to a non existing group will result in the group automatically being created.

Note that the master nodes are automatically assigned to be members of the pre-existing master group.

The following example configures the switch to be members of three groups; two are company departments, and one comprises all devices located in building_2. To avoid having to run this command separately on each device that is to be added to these groups, you can remotely assign all of these devices to a working-set, then use the capabilities of the working-set to apply the **atmf group (membership)** command to all members of the working set.

Example To specify the switch to become a member of AMF groups named, Marketing, Sales, and Building_2, use the following command:

```
VCF-1# configure terminal
VCF-1(config)# atmf group marketing,sales,building_2
```

First add the nodes “master_node1” and “member_node_1” to the working-set:

```
master_node# atmf working-set master_node1,member_node_1
```

This command returns the following output confirming that the nodes “master_node” and “node_2” are now part of the working-set:

```
====================
master_node1, member_node_1
======================

Working set join
```

```
atmf-net[2]# configure terminal
```

Add the groups building1 and sales to the working-set

```
atmf-net[2](config)# atmf group building1,sales
atmf-net[2](config)# exit
```

Show the groups that are members of the working-set

```
atmf-net[2]# show atmf group
```

This command returns the following output displaying the groups that are members of the working-set.

```
====================
master_node1
======================

AMF group information

building1, sales, master, poe, x8100
```

Related Commands **show atmf group**
show atmf group members

atmf log-verbose

This command limits the number of log messages displayed on the console or permanently logged.

Syntax

```
atmf log-verbose <1-3>
no atmf log-verbose
```

Parameter	Description
<1-3>	The verbose limitation (3 = noisiest, 1 = quietest)

Default The default log display is 3.

Usage This command is intended for use in large networks where verbose output can make the console unusable for periods of time while nodes are joining and leaving.

Mode Global Configuration

Example To set the log-verbose to noise level 2, use the command:

```
VCF-1# configure terminal
VCF-1(config)# atmf log-verbose 2
```

Validation Command **show atmf**

atmf management subnet

This command is used to assign a subnet that will be allocated to the AMF management and domain management VLANs. From the address space defined by this command, two subnets are created, a management subnet component and a domain component, as explained in the Usage section of this command description.

AMF uses these internal IPv4 subnets when exchanging its inter nodal status packets. These subnet addresses must be reserved for AMF and should be used for no other purpose.

The new management subnet will not become effective until all members of the AMF network have been updated and all its units rebooted.

Use the **no** variant of this command to remove the assigned subnet VLANs.

Syntax

```
atmf management subnet <a.b.0.0>

no atmf management subnet
```

Parameter	Description
<a.b.0.0>	The IP address selected for the management subnet. Because a mask of 255.255.0.0 (i.e. /16) will be applied automatically, an IP address in the format a.b.0.0 must be selected. Usually this subnet address is selected from an appropriate range from within the private address space of 172.16.0 to 172.31.255.255, or 192.168.0.0 as defined in RFC1918.

Default 172.31.0.0 (Note that a subnet mask of 255.255.0.0 will automatically be applied).

Mode Global Configuration

Usage Typically a network administrator would use this command to change the default subnet address to match local network requirements.

As previously mentioned, running this command will result in the creation of a further two subnets (within the class B address space assigned) and the mask will extend from /16 to /17.

For example, if the management subnet is assigned the address 172.31.0.0/16, this will result in the automatic creation of the following two subnets:

- 172.31.0.0/17 assigned to the **atmf management vlan**
- 172.31.128.0/17 assigned to the **atmf domain vlan**.

Examples To change the AMF management subnet address on node VCF-1 to 172.25.0.0:

```
VCF-1# configure terminal
VCF-1(config)# atmf management subnet 172.25.0.0
```

To change the AMF management subnet address on node VCF-1 back to its default of 172.31.0.0:

```
VCF-1# configure terminal
VCF-1(config)# no atmf management subnet
```

atmf management vlan

The AMF management VLAN is created when the AMF network is first initiated and is assigned its default VID of 4092. This command enables you to change the VID from this default value.

The AMF management vlan is one of the internal VLANs that are used to communicate information about the state of the AMF network between nodes. AMF uses its internal VLANS (such as the management VLAN and the domain VLAN) to communicate its inter nodal network status information. These VLANs must be reserved for AMF and not used for other purposes.

If you assign a VLAN ID to this VLAN (i.e. change its value from the default of 4092) then you will need to do this separately on every device within the AMF. The AMF management subnet will then be applied to this new VID when all devices within the AMF network are next rebooted.

Use the **no** variant of this command to restore the VID to the default of 4092.

Syntax

```
atmf management vlan <2-4090>

no atmf management vlan
```

Parameter	Description
<2-4090>	The VID assigned to the AMF management VLAN.

Default The default VLAN ID for the AMF is 4092.

Note Although the value applied by default lies outside the user configurable range. You can use the "no" form of this command to reset the VLAN to its default value.

Mode Global Configuration

Usage You can use this command to change the management VLAN to meet your network's requirements and standards, particularly in situations where the default address value is unacceptable.

Note This VLAN will automatically be assigned an IP subnet address based on the value configured by the command **"atmf management subnet" on page 86.21**. Refer to this command description for further details.

Examples To change the AMF management VLAN to 4090 use the following commands:

```
VCF-1# configure terminal
VCF-1(config)# atmf management vlan 4090
```

To reset the AMF domain VLAN to its default of 4092, use the following commands:

```
VCF-1# configure terminal
VCF-1(config)# no atmf management vlan
```

Related Commands **atmf domain vlan**
show atmf

atmf master

This command configures the switch to be an AMF master node and automatically creates an AMF master group. The master node is considered to be the core of the AMF network, and must be present for the AMF to form. The AMF master has its node depth set to 0. Note that the node depth vertical distance is determined by the number of uplinks/ downlinks that exist between the node and its master.

An AMF master node must be present for an AMF network to form. Up to two AMF master nodes may exist in a network, and they **must** be connected by an AMF crosslink.

Note
Master nodes are an essential component of an AMF network. In order to run AMF, an AMF License is required for each master node.

If the crosslink between two AMF masters fails, then one of the masters will become isolated from the rest of the AMF network.

Use the **no** variant of this command to remove the switch as an AMF master node. The node will retain its node depth of 0 until the network is rebooted.

Note
Node depth is the vertical distance (or level) from the master node (whose depth value is 0).

Syntax

```
atmf master
no atmf master
```

Default The switch is not configured to be an AMF master node.

Mode Global Configuration

Example To specify that this node is an AMF master, use the following command:

```
VCF-1# configure terminal
VCF-1(config)# atmf master
```

Related Commands

show atmf
show atmf group

atmf network-name

This command applies an AMF network name to a (prospective) AMF node. In order for an AMF network to be valid, its network-name must be configured on at least two nodes, one of which must be configured as a master and have an AMF License applied. These nodes may be connected using either AMF downlinks or crosslinks.

For more information on configuring an AMF master node, see **"atmf master" on page 86.25**.

Use the **no** variant of this command to remove the AMF network name.

Syntax

```
atmf network-name <name>

no atmf network-name
```

Parameter	Description
<name>	The AMF network name. Up to 15 printable characters can be entered for the network-name.

Mode Global Configuration

Usage This is one of the essential commands when configuring AMF and must be entered on each node that is to be part of the AMF. This command will not take effect until the particular node is rebooted.

A switching node (master or member) may be a member of only one AMF network.

Caution

Ensure that you enter the correct network name. Entering an incorrect name will cause the AMF network to fragment (at the next reboot).

Example To set the AMF network name to amf_net use the command:

```
Node_1(config)# atmf network-name amf_net
```

atmf provision

This command configures a specified port on an AMF node to accept a provisioned node, via an AMF link, some time in the future.

Use the **no** variant of this command to remove the provisioning on the node.

Syntax

```
atmf provision [<nodename>]
no atmf provision
```

Parameter	Description
<nodename>	The name of the provisioned node that will appear on the AMF network in the future.

Default No provision.

Mode Interface Configuration

Usage The port should be configured as an AMF link or cross link and should be "down" to add or remove a provisioned node.

Example To provision an AMF node named node1 for port1.0.1, use the command:

```
host1(config)# interface port1.0.1
host1(config-if)# atmf provision node1
```

Related Commands

switchport atmf-link
switchport atmf-crosslink
show atmf links

atmf provision node clone

This command sets up a space on the backup media for use with a provisioned node and copies into it almost all files and directories from a chosen backup or provisioned node.

Alternatively, you can set up a new, unique provisioned node by using the command **atmf provision node create**.

Syntax `atmf provision node <nodename> clone <source nodename>`

Parameter	Description
<nodename>	The name that will be assigned to the clone when connected.
<source nodename>	The name of the node whose configuration is to be copied for loading to the clone.

Mode Privileged Exec

Usage This command is only available on master nodes in the AMF network.

You must run either this command or **atmf provision node create** command, before you can use other "atmf provision node" commands using the specified node name. If a backup or provisioned node already exists for the specified node then you must delete it before using the **atmf provision node** clone command.

When using this command it is important to be aware of the following:

- A copy of <*media*>:atmf/<*atmf_name*>/nodes/<*source_node*>/flash will be made for the provisioned node and stored in the backup media.
- The directory <*node_backup_dir*>/flash/.config/ssh is excluded from the copy.
- All contents of <*root_backup_dir*>/nodes/<*nodename*> will be deleted or overwritten.
- Settings for the expected location of other provisioned nodes are excluded from the copy.

The active and backup configuration files are automatically modified in the following ways:

- The "hostname" command is modified to match the name of the provisioned node.
- The "stack virtual-chassis-id" command is removed, if present.

Example To copy from the backup of Switch2 to create backup files for the new provisioned node Switch3 use the following command:

```
switch1# atmf provision node switch3 clone switch2
```

Figure 86-1: Sample output from the atmf provision node clone command

```
switch1#atmf provision node switch3 clone switch2
Copying...
Successful operation
```

To confirm that a new provisioned node has been cloned, use the command:

```
switch1# show atmf backup
```

The output from this command is shown in **Figure 86-2**, below, and shows the details of the new provisioned node switch3.

Figure 86-2: Sample output from the show atmf backup command

```
switch1#show atmf backup

Scheduled Backup ...... Enabled
  Schedule ............ 1 per day starting at 03:00
  Next Backup Time .... 01 Jan 2014 03:00
Backup Bandwidth ...... Unlimited
Backup Media .......... USB (Total 7446.0MB, Free 7297.0MB)
Server Config .........
  Synchronization ..... Unsynchronized
    Last Run .......... -
   1 ................... Unconfigured
   2 ................... Unconfigured
Current Action ........ Idle
  Started ............. -
  Current Node ........ -

-----------------------------------------------------------------
Node Name     Date         Time      In ATMF  On Media  Status
-----------------------------------------------------------------
switch3       -            -         No       Yes       Prov
switch1       01 Jan 2014  00:05:49  No       Yes       Good
switch2       01 Jan 2014  00:05:44  Yes      Yes       Good
```

atmf provision node configure boot config

This command sets the configuration file to use during the next boot cycle. This command can also set a backup configuration file to use if the main configuration file cannot be accessed for an AMF provisioned node. To unset the boot configuration or the backup boot configuration use the no boot command.

Use the **no** variant of this command to set back to the default.

Syntax

```
atmf provision node <nodename> configure boot config [backup]
   [<file-path|URL>]
atmf provision node [<nodename>] configure no boot config [backup]
```

Parameter	Description
<nodename>	The name of the provisioned node.
<file-path|URL>	The path or URL and name of the configuration file.

Default No boot configuration files or backup configuration files are specified for the provisioned node.

Mode Privileged Exec

Usage When using this command to set a backup configuration file, the specified AMF provisioned node must exist. The specified file must exist in the flash directory created for the provisioned node in the AMF remote backup media.

Examples To set the configuration file `branch.cfg` on the AMF provisioned node `node1`, use the command:

```
MasterNodeName# atmf provision node node1 configure boot
config branch.cfg
```

To set the configuration file `backup.cfg` as the backup to the main configuration file on the AMF provisioned node `node1`, use the command:

```
MasterNodeName# atmf provision node node1 configure boot
config backup usb:/atmf/amf_net/nodes/node1/
config/backup.cfg
```

To unset the boot configuration, use the command:

```
MasterNodeName# atmf provision node node1 configure no boot
config
```

To unset the backup boot configuration, use the command:

```
MasterNodeName# atmf provision node node1 configure no boot
config backup
```

Related Commands **atmf provision node configure boot system**
show atmf provision nodes

atmf provision node configure boot system

This command sets the release file that will load onto a specified provisioned node during the next boot cycle. This command can also set the backup release file to be loaded for an AMF provisioned node. To unset the boot system release file or the backup boot release file use the no boot command.

Use the **no** variant of this command to set back to the default.

This command can only be run on amf master nodes.

Syntax

```
atmf provision node <nodename> configure boot system [backup]
   [<file-path|URL>]

atmf provision node <nodename> configure no boot system [backup]
```

Parameter	Description
<nodename>	The name of the provisioned node.
<file-path|URL>	The path or URL and name of the release file.

Default No boot release file or backup release files are specified for the provisioned node.

Mode Privileged Exec

Usage When using this command to set a backup release file, the specified AMF provisioned node must exist. The specified file must exist in the flash directory created for the provisioned node in the AMF remote backup media.

Examples To set the release file `x900-5.4.4-1.rel` on the AMF provisioned node `node1`, use the command:

```
MasterNodeName# atmf provision node node1 configure boot
system x900-5.4.4-1.rel
```

To set the backup release file `900-5.4.4-1.rel` as the backup to the main release file on the AMF provisioned node `node1`, use the command:

```
MasterNodeName# atmf provision node node1 configure boot
system backup card:/atmf/amf_net/nodes/
node1/flash/x900-5.4.4-1.rel
```

To unset the boot release, use the command:

```
MasterNodeName# atmf provision node node1 configure no boot
system
```

To unset the backup boot release, use the command:

```
MasterNodeName# atmf provision node node1 configure no boot
system backup
```

Related Commands **atmf provision node configure boot config**
show atmf provision nodes

atmf provision node create

This command sets up an empty directory on the backup media for use with a provisioned node. This directory can have configuration and release files copied to it from existing switches. Alternatively, the configuration files can be created by the user.

An alternative way to create a new provisioned node is with the command **atmf provision node clone**.

This command can only run on amf master nodes.

Syntax `atmf provision node <nodename> create`

Parameter	Description
<nodename>	The name of the node that is being provisioned.

Mode Privileged Exec

Usage This command is only available on master nodes in the AMF network.

The **atmf provision node create** **atmf** command (or **atmf provision node clone**) must be executed before you can use other "atmf provision node" commands with the specified node name. If a backup or provisioned node already exists for the specified node name then you must delete it before using this command.

A date and time is assigned to the new provisioning directory reflecting when this command was executed. If there is a backup or provisioned node with the same name on another AMF master then the most recent one will be used.

Example To create a new provisioned node named switch2 use the command:

```
switch1# atmf provision node switch2 create
```

Running this command will create the following directories:

- <*media*>:atmf/<*atmf_name*>/nodes/<*node*>
- <*media*>:atmf/<*atmf_name*>/nodes/<*node*>/flash

To confirm the new node's settings, use the command:

```
switch1# show atmf backup
```

The output for the show atmf backup command is shown in **Figure 86-3**, and shows details for the new provisioned node switch2.

Figure 86-3: Sample output from the show atmf backup command

```
switch1#show atmf backup

Scheduled Backup ...... Enabled
  Schedule ............ 1 per day starting at 03:00
  Next Backup Time .... 02 Jan 2014 03:00
Backup Bandwidth ...... Unlimited
Backup Media .......... USB (Total 7446.0MB, Free 7315.2MB)
Server Config .........
  Synchronization ..... Unsynchronized
    Last Run .......... -
   1 .................. Unconfigured
   2 .................. Unconfigured
Current Action ........ Idle
  Started ............. -
  Current Node ........ -

-----------------------------------------------------------------
Node Name     Date         Time      In ATMF  On Media  Status
-----------------------------------------------------------------
switch2       -            -         No       Yes       Prov
switch1       01 Jan 2014  00:05:49  No       Yes       Good
```

For instructions on how to configure on a provisioned node refer to **"Configuring AMF" on page 85.12** and **"Creating and Using Configuration Files" on page 6.8**.

Related commands **atmf provision node clone**

atmf provision node delete

This command deletes files that have been created for loading onto a provisioned node. It can only be run on master nodes.

Syntax `atmf provision node <nodename> delete`

Parameter	Description
<nodename>	The name of the provisioned node to be deleted.

Mode Privileged Exec

Usage This command is only available on master nodes in the AMF network. The command will only work if the provisioned node specified in the command has already been set up (although the device itself is still yet to be installed). Otherwise, an error message is shown when the command is run.

You may want to use the **atmf provision node delete** command to delete a provisioned node that was created in error or that is no longer needed.

This command cannot be used to delete backups created by the AMF backup procedure. In this case, use the command **atmf backup delete** to delete the files.

Note This command allows provisioned entries to be deleted even if they have been referenced by the **atmf provision** command, so take care to only delete unwanted entries.

Example To delete backup files for a provisioned node named switch3 use the command:

```
switch1# atmf provision node switch3 delete
```

To confirm that the backup files for provisioned node switch3 have been deleted use the command:

```
switch1# show atmf backup
```

The output should show that the provisioned node switch3 no longer exists in the backup file, as shown in **Figure 86-4**:

Figure 86-4: Sample output showing the show atmf backup command

```
switch1#show atmf backup

Scheduled Backup ...... Enabled
  Schedule ............ 1 per day starting at 03:00
  Next Backup Time .... 01 Jan 2014 03:00
Backup Bandwidth ...... Unlimited
Backup Media .......... USB (Total 7446.0MB, Free 7297.0MB)
Server Config ......
  Synchronization ..... Unsynchronized
    Last Run .......... -
   1 ................... Unconfigured
   2 ................... Unconfigured
Current Action ........ Idle
  Started ............. -
  Current Node ........ -

-----------------------------------------------------------------
Node Name     Date         Time      In ATMF  On Media  Status
-----------------------------------------------------------------
switch1       01 Jan 2014  00:05:49  No       Yes       Good
switch2       01 Jan 2014  00:05:44  Yes      Yes       Good
```

Related commands atmf provision node create

atmf provision node license-cert

This command is used to set up the license certificate for a provisioned node.

The certificate file usually has all the license details for the network, and can be stored anywhere in the network. This command makes a hidden copy of the certificate file and stores it in the space set up for the provisioned node on AMF backup media.

For node provisioning, the new device has not yet been part of the AMF network, so the user is unlikely to know its product ID or its MAC address. When such a device joins the network, assuming that this command has been applied successfully, the copy of the certificate file will be applied automatically to the provisioned node.

Once the new device has been resurrected on the network and the certificate file has been downloaded to the provisioned node, the hidden copy of the certificate file is deleted from AMF backup media.

Use the **no** variant of this command to set it back to the default.

This command can only be run on amf master nodes.

Syntax

```
atmf provision node {<nodename>} license-cert <file-path|URL>
no atmf provision node {<nodename>} license-cert
```

Parameter	Description
<nodename>	The name of the provisioned node.
<file-path|URL>	The name of the certificate file. This can include the file-path of the file.

Default No license certificate file is specified for the provisioned node.

Mode Privileged Exec

Usage This command is only available on master nodes in the AMF network. It will only operate if the provisioned node specified in the command has already been set up, and if the license certification is present in the backup file. Otherwise, an error message is shown when the command is run.

Example 1 To apply the license certificate cert1.txt stored on tftp server for AMF provisioned node “Switch2”, use the command:

```
switch1# atmf provision node switch2 license-cert
         tftp://192.168.1.1/cert1.txt
```

Example 2 To apply the license certificate cert2.txt stored on AMF master's flash directory for AMF provisioned node “host2”, use the command:

```
switch1# atmf provision node switch2 license-cert/
         cert2.txt
```

To confirm that the license certificate has been applied to the provisioned node, use the command **show atmf provision nodes**. The output from this command is shown **Figure 86-5** below, and displays license certification details in the last line.

Figure 86-5: Sample output from the show atmf provision nodes command

```
switch1#show atmf provision nodes

ATMF Provisioned Node Information:

Backup Media .............: SD (Total 3827.0MB, Free 3481.1MB)

Node Name                 : switch2
Date & Time               : 06-May-2014 & 23:25:44
Provision Path            : card:/atmf/nodes

Boot configuration :
Current boot image        : x510-1766_atmf_backup.rel (file exists)
Backup  boot image        : x510-main-20140113-2.rel (file exists)
Default boot config       : flash:/default.cfg (file exists)
Current boot config       : flash:/abc.cfg (file exists)
Backup  boot config       : flash:/xyz.cfg (file exists)

Software Licenses :
Repository file           : ./.configs/.sw_v2.lic
                          : ./.configs/.swfeature.lic
Certificate file          : card:/atmf/lok/nodes/awplus1/flash/.atmf-lic-cert
```

Related commands show atmf provision nodes

atmf provision node locate

This command changes the present working directory to the directory of a provisioned node. This makes it easier to edit files and create a unique provisioned node in the backup.

This command can only be run on amf master nodes.

Syntax `atmf provision node <nodename> locate`

Parameter	Description
<nodename>	The name of the provisioned node.

Mode Privileged Exec

Usage This command is only available on master nodes in the AMF network.The command will only work if the provisioned node specified in the command has already been set up. Otherwise, an error message is shown when the command is run.

Note We advise that after running this command, you return to a known working directory, typically flash.

Example To change the working directory that happens to be on switch1 to the directory of provisioned node switch2, use the following command:

```
switch1# atmf provision node switch2 locate
```

The directory of the node switch2 should now be the working directory. You can use the command **pwd** to check this, as shown in the following figure.

Figure 86-6: Sample output from the pwd command

```
switch2#pwd
card:/atmf/building_2/nodes/switch2/flash
```

The output above shows that the working directory is now the flash of switch2.

Related commands
atmf provision node create
atmf provision node clone
pwd

atmf reboot-rolling

This command enables you to reboot the nodes in an AMF working-set, one at a time, as a rolling sequence in order to minimize downtime. Once a rebooted node has finished running its configuration and its ports are up, it re-joins the AMF network and the next node is rebooted.

By adding the url parameter, you can also upgrade your switches' software one AMF node at a time.

The force command enforces a node reboot even if a previous node does not rejoin the AMF network. In this situation the unsuitable node will time-out and the rolling reboot process stops. However, with the **force** parameter applied, the process will ignore the timeout and move on to reboot the next node in the sequence.

This command can take a significant amount of time to complete.

Syntax `atmf reboot-rolling [force] [<url>]`

Parameter	Description
`force`	Ignore a failed node and move on to the next node. Where a node fails to reboot a timeout is applied based on the time taken during the last reboot.
<url>	The url path to the software upgrade file.

Mode Privileged Exec

Usage You can load the software from a variety of locations. The latest compatible release for a node will be selected from your selected location - based on the parameters and URL you have entered.

For example card:/5.4.3/x*-5.4.3-*.rel will select from the folder card:/5.4.3 the latest file that matches the selection x (wildcard) -5.4.3-(wildcard).rel. Because x* is applied, each switch type will be detected and its appropriate release file will be installed.

Other allowable entries are:

- card:*.rel:
  Used when loading SW from SD cards.
- tftp:ip address:
  Used when loading SW from a TFTP server.
- usb:
  Used when loading SW from a USB flash drive.
- flash:
  Used when loading SW from flash memory, i.e. from one x900 switch to another.
- scp:
  Used when loading SW from a secure copy.
- http:
  Used when loading SW from an HTTP file server site.

Several checks are performed to ensure the upgrade will succeed. These include checking the current node release boots from flash. If there is enough space on flash, the software release is copied to flash to a new location on each node as it is processed. The new release name will be updated using the "boot system *<release-name>*" command, and the old release will become the backup release file.

Note If you are using TFTP or HTTP, for example, to access a file on a remote device then the URL should specify the exact release filename without using wild card characters.

On bootup the software release is verified. Should an upgrade fail, the upgrading unit will revert back to its previous software version. At the completion of this command, a report is run showing the release upgrade status of each node.

Note Take care when removing external media or rebooting your switches. Removing an external media while files are being written entails a significant risk of causing a file corruption.

Example 1 To reboot all x510 nodes in an AMF network, use the following command:

```
Bld2_Floor_1# atmf working-set group x510
```

This command returns the following type of screen output:

```
====================
node1, node2, node3:
====================

Working set join

AMF_NETWORK_Name[3]#
```

```
ATMF_NETWORK[3]# atmf reboot-rolling
```

When the reboot has completed, a number of status screens appear. The selection of these screens will depend on the parameters set.

```
Bld2_Floor_1#atmf working-set group x510

============================
SW_Team1, SW_Team2, SW_Team3:
============================

Working set join


ATMF_NETWORK[3]#atmf reboot-rolling
ATMF Rolling Reboot Nodes:

                     Timeout
Node Name            (Minutes)
--------------------------
SW_Team1                14
SW_Team2                 8
SW_Team3                 8
Continue the rolling reboot ? (y/n):y
================================================
ATMF Rolling Reboot: Rebooting SW_Team1
================================================

% SW_Team1 has left the working-set
Reboot of SW_Team1 has completed
================================================
ATMF Rolling Reboot: Rebooting SW_Team2
================================================

% SW_Team2 has left the working-set
Reboot of SW_Team2 has completed
================================================
ATMF Rolling Reboot: Rebooting SW_Team3
================================================

% SW_Team3 has left the working-set
Reboot of SW_Team3 has completed

================================================
ATMF Rolling Reboot Complete
Node Name           Reboot Status
---------------------------------
SW_Team1               Rebooted
SW_Team2               Rebooted
SW_Team3               Rebooted
================================================
```

Example 2 To update firmware releases, use the following command:

```
Node_1# atmf working-set group all
ATMF_NETWORK[9]# atmf reboot-rolling card:/5.4.3/x*-5.4.3-*.rel
```

.

```
ATMF Rolling Reboot Nodes:

                    Timeout
Node Name           (Minutes)    New Release File             Status
-------------------------------------------------------------------
SW_Team1               8         x510-5.4.3-0.5.rel           Release Ready
SW_Team2              10         x510-5.4.3-0.5.rel           Release Ready
SW_Team3               8         ---                          Not Supported
HW_Team1               6         ---                          Incompatible
Bld2_Floor_1           6         x900-5.4.3-0.5.rel           Release Ready
Bld1_Floor_2           2         x610-5.4.3-0.5.rel           Release Ready
Bld1_Floor_1           4         ---                          Incompatible
Building_1             2         ---                          Incompatible
Building_2             2         x900-5.4.3-0.5.rel           Release Ready
Continue upgrading releases ? (y/n):
```

atmf recover

This command is used to manually initiate the recovery (or replication) of an AMF node, usually when a node is being replaced. The recovery/replication process involves loading the configuration file for a node that is either about to be replaced or has experienced some problem. The configuration file of the device being replaced is selected by the nodename parameter, and the master node holding the configuration file is specified by the parameter <*master-nodename*>.

If the <*nodename*> parameter is not entered then the node will attempt to use one that has been previously configured. If the replacement node has no previous configuration (and has no previously used nodename), then the recovery will fail.

If the <*master-nodename*> parameter is not specified then the device will poll all known AMF masters and execute an election process (based on the last successful backup and its timestamp) to determine which master node to use. If no valid backup master is found, then this command will fail.

Syntax `atmf recover [<nodename> <master-nodename>]`

Parameter	Description
`<nodename>`	The name of the device whose configuration is to be recovered or replicated.
`<master-nodename>`	The name of the master device that holds the required configuration information. Note that although you can omit both the nodename and the master nodename; you can only omit the master nodename if you also omit the nodename.

Mode Privileged Exec

Usage No error checking occurs when this command is run, and regardless of the last backup status, the recovering node will attempt to load its configuration from the master node specified by the master-nodename parameter.

Note that if the node has previously been configured, we recommend that you suspend any AMF backup before running this command. This is to prevent corruption of the backup files on the AMF master as it attempts to both backup and recover the node at the same time.

Example To recover the AMF node named Node_10 from the AMF master node named Master_2, use the following command:

```
Master_2# atmf recover Node_10 Master_2
```

Related Commands
atmf backup stop
show atmf backup
show atmf

atmf recover led-off

This command turns off the recovery failure flashing port LEDs. It reverts the LED's function to their normal operational mode, and in doing so assists with resolving the recovery problem. You can repeat this process until the recovery failure has been resolved. For more detailed information see **"Recovery progress indication" on page 85.36**.

Syntax `atmf recover led-off`

Default Normal operational mode

Mode Privileged Exec

Example To revert the LEDs on Node1 from recovery mode display, to their normal operational mode, use the command:

```
Node1# atmf recover led-off
```

Related Commands **atmf recover**

atmf remote-login

Use this command to remotely login to other AMF nodes in order to run commands as if you were a local user of that node.

Syntax `atmf remote-login [user <name>] <nodename>`

Parameter	Description
<name>	User name.
<nodename>	Node name.

Mode Privileged Exec (This command will only run at privilege level 15)

Usage You do not need a valid login on the local device in order to run this command. The session will take you to the enable prompt on the new device. If the remote login session exits for any reason (i.e. device reboot) you will be returned to the originating node.

The software will not allow you to run multiple remote login sessions. You must exit an existing session before starting a new one.

Example 1 To remotely login from node Node10 to Node20 use the following command:

```
Node10# atmf remote-login node20
```

Example 2 In this example, user Whitney is a valid user of node5. She can remotely login from node5 to node3 by using the following commands:

```
node5# atmf remote-login user whitney
       node3
Type 'exit' to return to node5#

node3> enable
```

Note In the above example the user name whitney is valid on both nodes.

Therefore, to prevent unauthorized access, user names should be unique across all nodes within the AMF network.

atmf restricted-login

This command restricts the use of the **"atmf working-set" on page 86.49** command on all AMF master nodes to privilege 15 users only. Once entered on any AMF master node, this command will propagate across the network.

Note that once you have run this command, certain other commands that utilize the AMF working-set command, such as the **include**, **atmf reboot-rolling** and **show atmf group members** commands, will operate only on master nodes.

Use the **no** variant of this command to disable restricted login on the AMF network. This allows access to the **atmf working-set** command from any node in the AMF network.

Syntax

```
atmf restricted-login

no atmf restricted-login
```

Mode Privileged Exec

Default Master nodes operate with **atmf restricted-login** disabled.

Member nodes operate with **atmf restricted-login** enabled.

Note

The default conditions of this command vary from those applied by its "no" variant. This is because the restricted-login action is only applied by **master** nodes, and in the absence of a master node, the default is to apply the restricted action to all **member** nodes with AMF configured.

In the presence of a **master** node, its default of "atmf restricted-login disabled" will permeate to all its member nodes. Similarly, any change in this command's status that is made on a master node, will also permeate to all its member nodes

Example To enable restricted login, use the command

```
Node_20(config)# atmf restricted-login node20
```

Validation Command **show atmf**

atmf virtual-link id ip remote-id remote-ip

This command creates one or more Layer 2 tunnels that enable AMF nodes to transparently communicate across a wide area network using Layer 2 connectivity protocols.

Once connected through the tunnel, the remote member will have the same AMF capabilities as a directly connected AMF member.

Use the **no** variant of this command to remove the specified virtual link.

Syntax

```
atmf virtual-link id <1-32> ip <a.b.c.d> remote-id <1-32> remote-ip
    <a.b.c.d>

no atmf virtual-link id <1-32>
```

Parameter	Description
`ip`	The Internet Protocol (IP).
<a.b.c.d>	The IP address, of the local amf node (at its interface to the tunnel) entered in a.b.c.d format.
`remote-id`	The ID of the (same) tunnel that will be applied by the remote node. Note that this must match the local-id that is defined on the remote node. This means that (for the same tunnel) the local and remote tunnel IDs are reversed on the local and remote nodes.
<1-32>	The ID range 1-32.
`remote-ip`	The IP address of the remote node
<a.b.c.d>	The IP address, of the remote node (at its interface to the tunnel) entered in a.b.c.d format.

Mode Privileged Exec

Usage The Layer 2 tunnel that this command creates enables a local AMF session to appear to pass transparently across a Wide Area Network (WAN) such as the Internet. The addresses configured as the local and remote tunnel IP addresses must have IP connectivity to each other. If the tunnel is configured to connect a head office and branch office over the Internet, typically this would involve using some type of managed WAN service such as a site-to-site VPN. Tunnels are only supported using IPv4.

Configuration involves creating a local tunnel ID, a local IP address, a remote tunnel ID and a remote IP address. A reciprocal configuration is also required on the corresponding remote device. The local tunnel ID must be unique to the device on which it is configured.

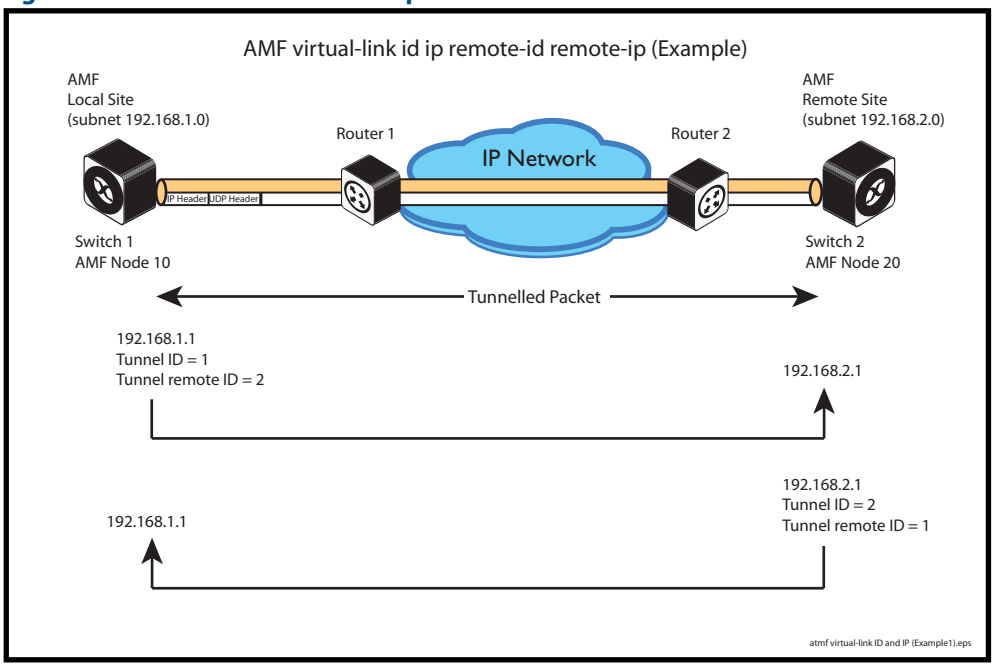
The tunneled link may operate via external (non AlliedWare Plus) routers in order to provide wide area network connectivity. However in this configuration, the routers perform a conventional router to router connection. The protocol tunneling function is accomplished by the AMF nodes.

Note The requirement to pre-configure the local IP address and tunnel ID on a device located at the far end of an AMF virtual-link tunnel means that zero touch device replacement cannot be achieved on a remote device that terminates the tunnel connection.

Example Use the following command to create the tunnel shown in figure **Figure 86-7 on page 86.48**.

```
Node_10(config)# atmf virtual-link id 1 ip 192.168.1.1
                 remote-id 2 remote-ip 192.168.2.1

Node_20(config)# atmf virtual-link id 2 ip 192.168.2.1
                 remote-id 1 remote-ip 192.168.1.1
```

Figure 86-7: AMF virtual link example



Validation Command **show atmf**

atmf working-set

The AMF working-set command enables you to execute commands across an individually listed set (or preselected group) of AMF nodes. Group selection is made using the **atmf group (membership)** command on page 86.18.

This command opens a session on multiple network devices. When you change the working set to anything other than the local device, the prompt will change to the AMF network name, followed by the size of the working set, shown in square brackets. This command has to be run at privilege level 15.

In addition to the user defined groups, the following system assigned groups are automatically created:

- Implicit Groups
  - « all - All nodes in the AMF
  - « current - All nodes that comprise the current working-set
  - « local - The originating node.
- Automatic Groups - These can be defined by hardware architecture, e.g. x510, x610, x900, x8100, or by certain AMF nodal designations such as master.

Note that the Implicit Groups do not appear in show group output.

If a node is an AMF master it will be automatically added to the master group.

Syntax

```
atmf working-set {[<node-list>][group{<group-list>|all|local|
   current}]}
```

Parameter	Description
<node-list>	A comma delimited list (without spaces) of nodes to be included in the working-set.
`group`	The AMF group.
<group-list>	A comma delimited list (without spaces) of groups to be included in the working-set. Note that this can include either defined groups, or any of the Automatic, or Implicit Groups shown earlier in the bulleted list of groups.
`all`	All nodes in the AMF.
`local`	Local node Running this command with the parameters **group local** will return you to the local prompt and local node connectivity.
`current`	Nodes in current list.

Default Needs to be entered

Mode Privileged Exec

Example 1 To add all nodes in the AMF to the working-set, use the command:

```
node1# atmf working-set group all
```

Note This command adds the implicit group “all” to the working set, where “all” comprises all nodes in the AMF.

This command displays an output screen similar to the one shown below:

```
=========================================
node1, node2, node3, node4, node5, node6:
=========================================

Working set join

ATMF_NETWORK_Name[6]#
```

Example 2 To return to the local prompt, and connectivity to only the local node; use the command:

```
ATMF_NETWORK_Name[6]# atmf working-set group local

                    node1#
```

Parameter	Description
`node1, node2`	The name of the nodes - as set by the **hostname** command on page 10.18.
`ATMF_Network_Name`	The name of the AMF network - as set by the **atmf network-name** command on page 86.26.
`[6]`	The number of nodes in the working-set.

clear atmf links statistics

This command resets the values of all AMF link, port, and global statistics to zero.

Syntax `clear atmf links statistics`

Mode Privilege Exec

Example To reset the AMF link statistics values, use the command:

```
node_1# clear atmf links statistics
```

Related Commands show atmf links statistics

debug atmf

This command enables the AMF debugging facilities, and displays information that is relevant (only) to the current node. The detail of the debugging displayed depends on the parameters specified.

If no additional parameters are specified, then the command output will display all AMF debugging information, including link events, topology discovery messages and all notable AMF events.

The "no" variant of this command disables either all AMF debugging information, or only the particular information as selected by the command's parameters.

Syntax

```
debug atmf [link|crosslink|database|neighbor|error|all]
no debug atmf [link|crosslink|database|neighbor|error|all]
```

Parameter	Description
`link`	Output displays debugging information relating to uplink or downlink information.
`crosslink`	Output displays all crosslink events.
`database`	Output displays only notable database events.
`neighbor`	Output displays only notable AMF neighbor events.
`error`	Output displays AMF error events.
`all`	Output displays all AMF events.

Default All debugging facilities are disabled.

Mode User Exec and Global Configuration

Usage If no additional parameters are specified, then the command output will display all AMF debugging information, including link events, topology discovery messages and all notable AMF events.

Note An alias to the no variant of this command is **"undebug atmf" on page 86.93**.

Examples To debug all AMF debugging, use the command:

```
node_1# debug atmf
```

To debug all AMF link debugging, use the command:

```
node_1# debug atmf link
```

To debug all AMF crosslink debugging, use the command:

```
node_1# debug atmf crosslink
```

To debug all AMF database debugging, use the command:

```
node_1# debug atmf database
```

To debug all AMF neighbor debugging, use the command:

```
node_1# debug atmf neighbor
```

To debug all AMF error debugging, use the command:

```
node_1# debug atmf error
```

To debug all AMF facilities, use the command:

```
node_1# debug atmf all
```

Related Commands **no debug all**

debug atmf packet

This command configures AMF Packet debugging parameters. The debug only displays information relevant to the current node. The command has following parameters:

Syntax

```
debug atmf packet[[direction {rx|tx|both}][level {1|2|3}][timeout
    <seconds>][num-pkts <quantity>][filter node <name> ][interface
    <ifname>][pkt-type {[1][2][3][4][5][6][7][8][9][10][11]]]
```

Simplified Syntax

debug atmf packet		[direction {rx|tx|both}]
		[level {[1][2|3}]
		[timeout <*seconds*>]
		[num-pkts <*quantity*>]
debug atmf packet	filter	[node <*name*>]
		[interface <*ifname*>]
		[pkt-type [1][2][3][4][5][6][7][8][9][10][11]]

Note You can combine the syntax components shown, but when doing so, you must retain their original order.

Default Level 1, both Tx and Rx, a timeout of 60 seconds with no filters applied.

Note An alias to the no variant of this command - **undebug atmf** - can be found elsewhere in this chapter.

Mode User Exec and Global Configuration

Usage If no additional parameters are specified, then the command output will apply a default selection of parameters shown below:

Parameter	Description
direction	Sets debug to packet received, transmitted, or both
rx	packets received by this node
tx	Packets sent from this node
1	AMF Packet Control header Information, Packet Sequence Number. Enter 1 to select this level.
2	AMF Detailed Packet Information. Enter 2 to select this level.
3	AMF Packet HEX dump. Enter 3 to select this level.
timeout	Sets the execution timeout for packet logging
<seconds>	Seconds
num-pkts	Sets the number of packets to be dumped

Parameter	Description
`pkts`	The actual number of packets
`filter`	Sets debug to filter packets
`node`	Sets the filter on packets for a particular Node
<name>	The name of the remote node
`interface`	Sets the filter to dump packets from an interface (portx.x.x) on the local node
`ifname`	Interface port or virtual-link
`pkt-type`	Sets the filter on packets with a particular AMF packet type
1	Crosslink Hello BPDU packet with crosslink links information. Enter 1 to select this packet type.
2	Crosslink Hello BPDU packet with downlink domain information. Enter 2 to select this packet type.
3	Crosslink Hello BPDU packet with uplink information. Enter 3 to select this packet type.
4	Downlink and uplink hello BPDU packets. Enter 4 to select this packet type.
5	Non broadcast hello unicast packets. Enter 5 to select this packet type.
6	Stack hello unicast packets. Enter 6 to select this packet type.
7	Database description. Enter 7 to select this packet type.
8	DBE request. Enter 8 to select this packet type.
9	DBE update. Enter 9 to select this packet type.
10	DBE bitmap update. Enter 10 to select this packet type.
11	DBE acknowledgment. Enter 11 to select this packet type.

Examples To set a packet debug on node 1 with level 1 and no timeout, use the command:

```
node_1# debug atmf packet direction tx timeout 0
```

To set a packet debug with level 3 and filter packets received from AMF node 1:

```
node_1# debug atmf packet direction tx level 3 filter
node_1
```

To enable send and receive 500 packets only on vlink1 for packet types 1, 7, and 11, use the command:

```
node_1# debug atmf packet num-pkts 500 filter interface
vlink1 pkt-type 1 7 11
```

This example applies the debug atmf packet command and combines many of its options:

```
node_1# debug atmf packet direction rx level 1 num-pkts
60 filter node x900 interface port1.0.1 pkt-
type 4 7 10
```

erase factory-default

This command erases all data from NVS and all data from flash **excluding** the following:

- The current release file and its /flash/.release file
- The backup release file and /flash/.backup file
- v1 license files /flash/.configs/.swfeature.lic
- v2 license files /flash/.configs/.sw_v2.lic

The device is then rebooted and returns the switch to its factory default condition. The switch can then be used for automatic node recovery.

Syntax `erase factory-default`

Mode Global Configuration.

Usage This command is an alias to the **atmf cleanup** command on page 86.13.

Example To erase data, use the command:

```
Node_1(config)# erase factory-default

This command will erase all NVS, all flash contents except
for the boot release, and any license files, and then
reboot the switch. Continue? (y/n):y
```

Related Commands **atmf cleanup**

show atmf

Displays information about the current AMF node.

Syntax `show atmf [summary|tech|nodes|session]`

Parameter	Description
`summary`	Displays summary information about the current AMF node.
`tech`	Displays global AMF information.
`nodes`	Displays a list of AMF nodes together with brief details.
`session`	Displays information on an AMF session.

Default Only summary information is displayed.

Mode User Exec and Privileged Exec

Usage AMF uses internal VLANs to communicate between nodes about the state of the AMF network. Two VLANs have been selected specifically for this purpose. Once these have been assigned, they are reserved for AMF and cannot be used for other purposes

Example 1 To show summary information on AMF node_1 use the following command:

```
node_1 show atmf summary
```

The following figure shows some example output from running this command for a specific AMF node.

Figure 86-8: Output from the show atmf summary command

```
node_1#show atmf
ATMF Summary Information:

ATMF Status            : Enabled
Network Name           : ATMF_NET
Node Name              : node_1
Role                   : Master
Current ATMF Nodes     : 8
```

Example 2 To show information specific to AMF nodes use the following command:

```
node_1 show atmf nodes
```

Figure 86-9: Output from the show atmf nodes command

```
Node Information:

  * = Local device

 SC = Switch Configuration:
  C = Chassis    S = Stackable    N = Standalone

  Node                Device             AMF                              Node
  Name                Type               Master  SC  Parent                Depth
-------------------------------------------------------------------------------
  Building_1          AT-SBx8112           Y     C   -                      0
* Building_2          x900-12XT/S          Y     N   -                      0
  Bld1_Floor_1        SwitchBlade x908     N     S   Building_1             1
  Bld1_Floor_2        x600-24Ts/XP         N     N   Building_1             1
  Bld2_Floor_1        x610-24Ts-POE+       N     N   Building_1             1
  SW_Team1            x510-28GPX           N     N   Bld1_Floor_2           2

 Current AMF node count 8
```

The **show amf session** command displays all CLI (Command Line Interface) sessions for users that are currently logged in and running a CLI session. For example, in the case below, node_1 and node5 have active users logged in.

Example 3 To display AMF active sessions, use the following command:

```
node_1 show atmf session
```

Figure 86-10: Output from the show atmf session command

```
node_1#show atmf session

CLI Session Neighbors

Session ID                            : 73518
Node Name                             : node_1
PID                                   : 7982
Link type                             : Broadcast-cli
MAC Address                           : 0000.0000.0000
Options                               : 0
Our bits                              : 0
Link State                            : Full
Domain Controller                     : 0
Backup Domain Controller              : 0
Database Description Sequence Number  : 00000000
First Adjacency                       : 1
Number Events                         : 0
DBE Retransmit Queue Length           : 0
DBE Request List Length               : 0

Session ID                            : 410804
Node Name                             : node5
PID                                   : 17588
Link type                             : Broadcast-cli
MAC Address                           : 001a.eb56.9020
Options                               : 0
Our bits                              : 0
Link State                            : Full
Domain Controller                     : 0
Backup Domain Controller              : 0
Database Description Sequence Number  : 00000000
First Adjacency                       : 1
Number Events                         : 0
DBE Retransmit Queue Length           : 0
DBE Request List Length               : 0
```

The AMF tech command collects all the AMF commands, and displays them. You can use this command when you want to see an overview of the AMF network.

Example 4 To display AMF technical information, use the following command:

```
node_1 show atmf tech
```

Figure 86-11: Output from the show atmf tech command

```
node_1#show atmf tech
ATMF Summary Information:

ATMF Status            : Enabled
Network Name           : ATMF_NET
Node Name              : node_1
Role                   : Master
Current ATMF Nodes     : 8

ATMF Technical information:

Network Name                          : ATMF_NET
Domain                                : node_1's domain
Node Depth                            : 0
Domain Flags                          : 0
Authentication Type                   : 0
MAC Address                           : 0014.2299.137d
Board ID                              : 287
Domain State                          : DomainController
Domain Controller                     : node_1
Backup Domain Controller              : node2
Domain controller MAC                 : 0014.2299.137d
Parent Domain                         : -
Parent Domain Controller              : -
Parent Domain Controller MAC          : 0000.0000.0000
Number of Domain Events               : 0
Crosslink Ports Blocking              : 0
Uplink Ports Waiting on Sync          : 0
Crosslink Sequence Number             : 7
Domains Sequence Number               : 28
Uplink Sequence Number                : 2
Number of Crosslink Ports             : 1
Number of Domain Nodes                : 2
Number of Neighbors                   : 5
Number of Non Broadcast Neighbors     : 3
Number of Link State Entries          : 1
Number of Up Uplinks                  : 0
Number of Up Uplinks on This Node     : 0
DBE Checksum                          : 84fc6
Number of DBE Entries                 : 0
Management Domain Ifindex             : 4391
Management Domain VLAN                : 4091
Management ifindex                    : 4392
Management VLAN                       : 4092
```

Table 86-1: Parameter definitions from the show atmf tech command

Parameter	Definition
ATMF Status	The Node's AMF status, either Enabled or Disabled.
Network Name	The AMF network that a particular node belongs to.
Node Name	The name assigned to a particular node.
Role	The role configured for this AMF device, either Master or Member.
Current ATMF Nodes	The count of AMF nodes in an AMF Network.
Node Address	An Address used to access a remotely located node (.atmf).

Table 86-1: Parameter definitions from the show atmf tech command

Parameter	Definition
Node ID	A Unique identifier assigned to a Node on an AMF network.
Node Depth	The number of nodes in path from this node to level of the AMF root node. It can be thought of as the vertical depth of the AMF network from a particular node to the zero level of the AMF root node.
Domain State	The state of Node in a Domain in AMF network as Controller/Backup.
Recovery State	The AMF node recovery status. Indicates whether a node recovery is in progress on this device - Auto, Manual, or None.
Management VLAN	The VLAN created for traffic between Nodes of different domain (up/down links). ■ VLAN ID - In this example VLAN 4092 is configured as the Management VLAN. ■ Management Subnet - Network prefix for the subnet. ■ Management IP Address - The IP address allocated for this traffic. ■ Management Mask - The subnet mask used to create a subnet for this traffic (255.255.128.0).
Domain VLAN	The VLAN assigned for traffic between Nodes of same domain (crosslink). ■ VLAN ID - In this example VLAN 4091 is configured as the domain VLAN. ■ Domain Subnet. The subnet address used for this traffic. ■ Domain IP Address. The IP address allocated for this traffic. ■ Domain Mask. The subnet mask used to create a subnet for this traffic (255.255.128.0).
Device Type	The Product Series Name.
ATMF Master	The 'Y' if the node belongs to a Core domain.
SC	The Switch Configuration, C - Chassis(SBx81series), S - Stackable (VCS) and N - Standalone.
Parent	The a Node to which the current node has an active uplink.
Node Depth	The the number of nodes in path from this node to the Core domain.

Related Commands **show atmf detail**

Note

You can manage your show output, or make it a more selective, by using a command modifier. For information on using show-command modifiers see **"Controlling "show" Command Output" on page 1.37**.

show atmf backup

This command displays information about AMF backup status for all the nodes in an AMF network. It can only be run on AMF master nodes.

Syntax `show atmf backup [logs|server-status|synchronize [logs]]`

Parameter	Description
`logs`	Displays detailed log information.
`server-status`	Displays connectivity diagnostics information for each configured remote file server.
`synchronize`	Display the file server synchronization status
`logs`	For each remote file server, display the logs for the last synchronization

Mode Privileged Exec

Example 1 To display the AMF backup information, use the command:

```
node_1# show atmf backup
```

```
Node_1# show atmf backup
ScheduledBackup ......Enabled
  Schedule.............1 per day starting at 03:00
  Next Backup Time....19 May 2012 03:00
Backup Media..........SD (Total 1974.0 MB, Free197.6MB)
Current Action.........Starting manual backup
Started................18 May 2012 10:08
CurrentNode............atmf_testbox1

Node Name                          Date         Time      In ATMF     Status
-------------------------------------------------------------------------------
atmf_testbox1                       17May  2014  09:58:59  Yes         Errors
atmf_testbox2                       17May  2014  10:01:23  Yes         Good

Node_1#show atmf backup logs

Log File Location: card:/atmf/office/logs/rsync_<nodename>.log

Node
Name Log Details-------------------------------------------------------------------
atmf_testbox2
      2014/05/22 03:41:32 [30299]File list size: 6199
      2014/05/22 03:41:32 [30299]File list generation time: 0.011 seconds
      2014/05/22 03:41:32 [30299]File list transfer time: 0.000 seconds
      2014/05/22 03:41:32 [30299]Total bytes sent: 696
      2014/05/22 03:41:32 [30299]Total bytes received: 16.03K
      2014/02/20 03:41:32 [30299]sent 696 bytes rece ived 16.03Kbytes 33.45 K
      bytes/sec
      2014/05/22 03:41:32 [30299]total size is 21.73M speedup is 1298.93
      2014/05/22 03:41:32 [30297]sent 626 bytes received 6203 bytes total
      size 43451648
```

Example 2 To display the AMF backup information with the optional parameter `server-status`, use the command:

```
Node_1# show atmf backup server-status
```

```
Node1#sh atmf backup server-status

Id  Last Check  State
-------------------------------------
1       186 s  File server ready
2           1 s  SSH no route to host
```

Table 86-2: Parameter definitions from the show atmf backup server-status command

Parameter	Definition
`Scheduled Backup`	Indicates whether AMF backup scheduling is enabled or disabled.
`Schedule`	Displays the configured backup schedule.
`Next Backup Time`	Displays the date and time of the next scheduled.
`Backup Media`	The current backup medium in use. This will be one of USB, SD, or NONE. Note that the USB will take precedence over the SD card. Utilized and available memory (MB) will be indicated if backup media memory is present.
`Current Action`	The task that the AMF backup mechanism is currently performing. This will be a combination of either (Idle, Starting, Doing, Stopping), or (manual, scheduled).
`Started`	The date and time that the currently executing task was initiated in the format DD MMM YYYY.
`Current Node`	The name of the node that is currently being backed up.
`Node Name`	The name of the node that is storing backup data - on its backup media.
`Date`	The data of the last backup in the format DD MMM YYYY.
`Time`	The time of the last backup in the format HH:MM:SS.
`In ATMF`	Whether the node shown is active in the AMF network, (Yes or No).
`Status`	The output can contain one of four values: ■ "-" meaning that the status file cannot be found or cannot be read. ■ "Errors" meaning that there are issues - note that the backup may still be deemed successful depending on the errors. ■ "Stopped" meaning that the backup attempt was manually aborted;. ■ "Good" meaning that the backup was completed successfully.
`Log File Location`	All backup attempts will generate a result log file in the identified directory based on the node name. In the above example this would be: card:/amf/office/logs/rsync_amf_testbox1.log.
`Log Details`	The contents of the backup log file.
`server-status`	Displays connectivity diagnostics information for each configured remove file server.

Note

You can manage your show output, or make it a more selective, by using a command modifier. For information on using show-command modifiers see **"Controlling "show" Command Output" on page 1.37**.

Related Commands **show atmf**
atmf network-name

show atmf detail

This command displays details about an AMF node. It can only be run on AMF master nodes.

Syntax `show atmf [detail]`

Parameter	Description
`detail`	Displays output in greater depth.

Mode Privileged Exec

Example 1 To display the AMF node1 information in detail, use the command:

```
node1# show atmf detail
```

A typical output screen from this command is shown below:

```
node1#show atmf detail
ATMF Detail Information

Network Name                        : ATMF_NET
Node Name                           : Admin2
Node Address                        : Admin2.atmf
Node ID                             : 15
Node Depth                          : 0
Domain State                        : DomainController
Recovery State                      : None

Management VLAN
 VLAN ID                            : 4092
 Management Subnet                  : 172.31.0.0
 Management IP Address              : 172.31.0.1
 Management Mask                    : 255.255.128.0

Domain VLAN
 VLAN ID                            : 4091
 Domain Subnet                      : 172.31.128.0
 Domain IP Address                  : 172.31.128.1
 Domain Mask                        : 255.255.128.0
```

Table 86-3: Parameter definitions from the show atmf details command

Parameter	Definition
`ATMF Status`	The Node's AMF status, either Enabled or Disabled.
`Network Name`	The AMF network that a particular node belongs to.
`Node Name`	The name assigned to a particular node.
`Role`	The role configured for this AMF device, either Master or Member.
`Current ATMF Nodes`	The count of AMF nodes in an AMF Network.
`Node Address`	An Address used to access a remotely located node. This is simply the Node Name plus the dotted suffix atmf (.atmf).
`Node ID`	A Unique identifier assigned to a Node on an AMF network.
`Node Depth`	The number of nodes in path from this node to level of the AMF root node. It can be thought of as the vertical depth of the AMF network from a particular node to the zero level of the AMF root node.

Table 86-3: Parameter definitions from the show atmf details command

Parameter	Definition
Domain State	The state of Node in a Domain in AMF network as Controller/Backup.
Recovery State	The AMF node recovery status. Indicates whether a node recovery is in progress on this device - Auto, Manual, or None.
Management VLAN	The VLAN created for traffic between Nodes of different domain (up/down links). ■ VLAN ID - In this example VLAN 4092 is configured as the Management VLAN. ■ Management Subnet - Network prefix for the subnet. ■ Management IP Address - The IP address allocated for this traffic. ■ Management Mask - The subnet mask used to create a subnet for this traffic (255.255.128.0).
Domain VLAN	The VLAN assigned for traffic between Nodes of same domain (crosslink). ■ VLAN ID - In this example VLAN 4091 is configured as the domain VLAN. ■ Domain Subnet. The subnet address used for this traffic. ■ Domain IP Address. The IP address allocated for this traffic. ■ Domain Mask. The subnet mask used to create a subnet for this traffic (255.255.128.0).
Device Type	The Product Series Name.
ATMF Master	'Y' if the node belongs to a Core domain.
SC	The Switch Configuration, C - Chassis(SBx81series), S - Stackable (VCS) and N - Standalone.
Parent	The a Node to which the current node has an active uplink.
Node Depth	The number of nodes in the path from this node to the Core domain.

show atmf group

This command can be used to display the group membership within to a particular AMF node. It can also be used with the working-set command to display group membership within a working set.

Each node in the AMF is automatically added to the group that is appropriate to its hardware architecture, e.g. x510, x610. Nodes that are configured as masters are automatically assigned to the master group.

You can create arbitrary groups of AMF members based on your own selection criteria. You can then assign commands collectively to any of these groups.

Syntax `show atmf group [user-defined|automatic]`

Parameter	Description
`user-defined`	User-defined-group information display.
`automatic`	Automatic group information display.

Default All groups are displayed

Mode Privileged Exec

Example 1 To display group membership of node2, use the following command:

```
node2# show atmf group
```

A typical output screen from this command is shown below:

```
ATMF group information
master, x510
node2#
```

This screen shows that node2 contains the groups, master and x510. Note that although the node also contains the implicit groups, these do not appear in the show output.

Example 2 The following commands (entered on node2) will display all the automatic groups within the working set containing node-1 and all nodes that have been pre-defined to contain the sysadmin group:

First define the working-set:

```
Node-1# #atmf working-set node-1 group sysadmin
```

A typical output screen from this command is shown below:

```
ATMF group information

master, poe, x8100


================================================
node-1, node-2, node-3, node-4, node-5, node-6:
================================================

ATMF group information

sysadmin, x8100

ATMF-Test-NETWORK[6]#
```

This confirms that the six nodes (node-1 to node-6) are now members of the working-set and that these nodes reside within the AMF-Test-Network.

Note that to run this command, you must have previously entered the command **"atmf working-set" on page 86.49**. This can be seen from the network level prompt, which in this case is, ATM_Network{6}#.

Figure 86-12: Sample output from the show atmf group command for a working set.

```
ATMF_NETWORK[6]#show atmf group
==============================
node-3, node-4, node-5, node-6:
==============================

ATMF group information

edge_switches, x510
```

Table 86-4: Parameter definitions from the show atmf group command for a working set

Parameter	Definition
`ATMF group information`	Displays a list of nodes and the groups that they belong to, for example: ■ master - Shows a common group name for Nodes configured as AMF masters. ■ Hardware Arch - Shows a group for all Nodes sharing a common Hardware architecture, e.g. x8100, x900, x610, for example. ■ User-defined - Arbitrary groups created by the user for AMF nodes.

show atmf group members

This command will display all group memberships within an AMF working-set. Each node in the AMF working set is automatically added to automatic groups which are defined by hardware architecture, e.g. x510, x610. Nodes that are configured as masters are automatically assigned to the master group. User can define arbitrary groupings of AMF members based on their own criteria, which can be used to select groups of nodes.

Syntax `show atmf group members [user-defined|automatic]`

Parameter	Description
`user-defined`	User defined group membership display.
`automatic`	Automatic group membership display.

Mode Privileged Exec

Example To display group membership of all nodes in a working-set, use the command:

```
ATMF_NETWORK[9]# show atmf group members
```

Figure 86-13: Sample output from the show atmf group members command

```
ATMF Group membership


Automatic        Total
Groups           Members   Members
----------------------------------------------------------------
master           1         Building_1
poe              1         HW_Team1
x510             3         SW_Team1 SW_Team2 SW_Team3
x900             1         Bld1_Floor_2
x610             1         HW_Team1
x8100            2         Building_1 Building_2


ATMF Group membership

User-defined     Total
Groups           Members   Members
----------------------------------------------------------------
marketing          1       Bld1_Floor_1
software           3       SW_Team1 SW_Team2 SW_Team3
```

Table 86-5: Parameter definitions from the show atmf group members command

Parameter	Definition
`Automatic Groups`	Lists the Automatic Groups and their nodal composition. The sample output shows AMF nodes based on same Hardware type or belonging to same Master group.
`User-defined Groups`	Shows grouping of AMF nodes in user defined groups.
`Total Members`	Shows the total number of members in each group.
`Members`	Shows the list of AMF Nodes in each group.

Related Commands

show atmf group
show atmf
atmf group (membership)

show atmf links

This command displays brief information about AMF links on a switch, such as link status and adjacent nodes.

Provisioned node names will be displayed with a trailing * character, and will not have an entry under Adjacent Ifindex.

This command can only be run on AMF master nodes.

Syntax `show atmf links`

Mode User Exec and Privileged Exec

Example To display the AMF links brief details, use the following command:

```
switch1# show atmf links brief
```

Figure 86-14: Sample output from the show atmf links command

```
switch1# show atmf link brief

ATMF  Links Brief:

Local    Link      Port    ATMF         Adjacent             Adjacent Link
Port     Type      Status  State        Node                 Ifindex  State
-------------------------------------------------------------------------
sa1      Crosslink Up      TwoWay       Building_1           4501     Forwarding
1.0.1    Downlink  Up      Full         Bld1_Floor_1         5001     Forwarding
1.0.2    Downlink  Up      Full         Bld1_Floor_2         5003     Forwarding
1.0.3    Downlink  Up      Full         Bld2_Floor_1         6101     Forwarding
1.0.4    Crosslink Down    Init         *switch3                      Blocking

* = provisioned
```

Related Commands
- no debug all
- clear atmf links statistics
- show atmf
- show atmf nodes

show atmf links detail

This command displays detailed information on all the links configured in the AMF network. It can only be run on AMF master nodes.

Syntax `show atmf links [detail]`

Parameter	Description
`detail`	Detailed AMF links information.

Mode User Exec

Example To display the AMF link details use this command:

```
switch1# show atmf links detail
```

The output from this command will display all the internal data held for AMF links.

Figure 86-15: Sample output from the show atmf links detail command

```
switch1# show atmf links details

ATMF  Links Detail:

Port                                : sa1
Ifindex                             : 4501
VR ID                               : 0
Port Status                         : Up
Port State                          : Full
Port BPDU Receive Count             : 44441
Adjacent Node Name                  : Building_2
Adjacent Ifindex                    : 4501
Adjacent VR ID                      : 0
Adjacent MAC                        : 0014.2299.137d
Port Last Message Response          : 0
```

Figure 86-15: Sample output from the show atmf links detail command (cont.)

```
Port                                    : port2.0.2
Ifindex                                 : 6002
VR ID                                   : 0
Port Status                             : Down
Port State                              : Init
Port BPDU Receive Count                 : 0

Link State Entries:

Node.Ifindex                            : Building_2.4501 -
Building_1.4501
Transaction ID                          : 3 - 3
MAC Address                             : 0014.2299.137d -
eccd.6d03.10e3
Link State                              : Full - Full

Domain Nodes Tree:

Node                                    : Building_2
  Links on Node                         : 1
  Link 0                                : Building_2.4501 -
Building_1.4501
  Forwarding State                      : Forwarding
Node                                    : Building_1
  Links on Node                         : 1
  Link 0                                : Building_2.4501 -
Building_1.4501
  Forwarding State                      : Forwarding
```

```
Crosslink Transaction Entries:

Node                                    : Building_2
Transaction ID                          : 3
Uplink Transaction ID                   : 3
Uplink Information:

Waiting for Sync                        : 0
Transaction ID                          : 3
Number of Links                         : 0
Number of Local Uplinks                 : 0
```

```
Uplink Information:

Waiting for Sync                        : 0
Transaction ID                          : 3
Number of Links                         : 0
Number of Local Uplinks                 : 0

Originating Node                        : Building_2
Domain                                  : -'s domain
Node                                    : Building_2
Ifindex                                 : 0
VR ID                                   : 0
Transaction ID                          : 3
Flags                                   : 32
Domain Controller                       : -
Domain Controller MAC                   : 0000.0000.0000
```

Figure 86-15: Sample output from the show atmf links detail command (cont.)

```
Downlink Domain Information:

Domain                                : Bld2_Floor_1's domain
  Domain Controller                   : Bld2_Floor_1
  Domain Controller MAC               : eccd.6d3f.fef7
  Number of Links                     : 2
  Number of Links Up                  : 2
  Number of Links on This Node        : 1
  Links are Blocked                   : 0
  Node Transaction List
    Node                              : Building_2
    Transaction ID                    : 7
  Domain List
    Domain                            : Bld2_Floor_1's domain
      Node                            : Building_2
      Ifindex                         : 5002
      Transaction ID                  : 7
      Flags                           : 1

    Domain                            : Bld2_Floor_1's domain
      Node                            : Building_1
      Ifindex                         : 7002
      Transaction ID                  : 7
      Flags                           : 1
```

```
------------------------------------------------------------------
----------------
Up/Downlink Ports Information
------------------------------------------------------------------
----------------
Port                                  : port1.3.1
Ifindex                               : 7001
VR ID                                 : 0
Port Status                           : Up
Port State                            : Full
Adjacent Node                         : Bld1_Floor_1
Adjacent Internal ID                  : 4
Adjacent Ifindex                      : 6001
Adjacent Board ID                     : 290
Adjacent VR ID                        : 0
Adjacent MAC                          : 0000.cd37.0ea4
Adjacent Domain Controller            : Bld1_Floor_1
Adjacent Domain Controller MAC        : 0000.cd37.0ea4
Port Forwarding State                 : Blocking
Port BPDU Receive Count               : 0
Port Sequence Number                  : 12
Port Adjacent Sequence Number         : 9
Port Last Message Response            : 0
```

```
Port                                  : port1.3.2
Ifindex                               : 7002
VR ID                                 : 0
Port Status                           : Up
Port State                            : Full
Adjacent Node                         : Bld2_Floor_1
Adjacent Internal ID                  : 3
Adjacent Ifindex                      : 5001
Adjacent Board ID                     : 333
Adjacent VR ID                        : 0
Adjacent MAC                          : eccd.6d3f.fef7
Adjacent Domain Controller            : Bld2_Floor_1
Adjacent Domain Controller MAC        : eccd.6d3f.fef7
Port Forwarding State                 : Blocking
Port BPDU Receive Count               : 0
Port Sequence Number                  : 15
Port Adjacent Sequence Number         : 8
Port Last Message Response            : 0
```

Table 86-6: Parameter definitions from the show atmf links detail command output

Parameter	Definition
`Port Status`	Shows status of the local port on the Node as UP/DOWN.
`Adjacent Node`	Shows Adjacent AMF Node to this Node.
`Adjacent IfIndex`	Shows interface on the Adjacent AMF Node connected to this Node.
`Link State`	Shows state of AMF link Forwarding/Blocking.
`Crosslink Ports Information`	Show details of all Crosslink ports on this Node: ■ Port - Name of the Port or static aggregation (sa<*>). ■ Ifindex - Interface index for the crosslink port. ■ VR ID - Virtual router id for the crosslink port. ■ Port Status - Shows status of the local port on the Node as UP/DOWN. ■ Port State - Same as AMF state as described above. ■ Port BPDU Receive Count - The number of AMF protocol PDU's received. ■ Adjacent Node Name - name of the adjacent node in the domain. ■ Adjacent Ifindex - Ifindex of the adjacent node in the domain. ■ Adjacent VR ID - Virtual router id of the adjacent node in the domain. ■ Adjacent MAC - MAC address of the adjacent node in the domain. ■ Port Last Message Response - Response from the remote neighbor to our AMF last hello packet.
`Link State Entries`	Show all the link state database entries: ■ Node.Ifindex - Shows adjacent Node names and Interface index. ■ Transaction ID - Shows transaction id of the current crosslink transaction. ■ MAC Address - Shows adjacent Node MAC addresses. ■ Link State - Shows AMF states of adjacent nodes on the link.
`Domain Nodes Tree`	Shows all the nodes in the domain: ■ Node - Name of the node in the domain. ■ Links on Node - Number of crosslinks on a vertex/node. ■ Link no - Shows adjacent Node names and Interface index. ■ Forwarding State - Shows state of AMF link Forwarding/Blocking.
`Crosslink Transaction Entries`	Shows all the transaction entries: ■ Node - Name of the AMF node. ■ Transaction ID - transaction id of the node. ■ Uplink Transaction ID - transaction id of the remote node.
`Uplink Information`	Show all uplink entries. ■ Waiting for Sync - Flag if uplinks are currently waiting for synchronization. ■ Transaction ID - Shows transaction id of the local node. ■ Number of Links - Number of up downlinks in the domain. ■ Number of Local Uplinks - Number of uplinks on this node to the parent domain. ■ Originating Node - Node originating the uplink information. ■ Domain - Name of the parent uplink domain. ■ Node - Name of the node in the parent domain, that is connected to the current domain. ■ Ifindex - Interface index of the parent node's link to the current domain. ■ VR ID - Virtual router id of the parent node's link to the current domain. ■ Transaction ID - Transaction identifier for the neighbor in crosslink. ■ Flags - Used in domain messages to exchange the state: ■ATMF_DOMAIN_FLAG_DOWN = 0 ■ATMF_DOMAIN_FLAG_UP = 1 ■ATMF_DOMAIN_FLAG_BLOCK = 2 ■ATMF_DOMAIN_FLAG_NOT_PRESENT = 4 ■ATMF_DOMAIN_FLAG_NO_NODE = 8 ■ATMF_DOMAIN_FLAG_NOT_ACTIVE_PARENT = 16 ■ATMF_DOMAIN_FLAG_NOT_LINKS = 32 ■ATMF_DOMAIN_FLAG_NO_CONFIG = 64 ■ Domain Controller - Domain Controller in the uplink domain ■ Domain Controller MAC - MAC address of Domain Controller in uplink domain

Table 86-6: Parameter definitions from the show atmf links detail command output(cont.)

Parameter	Definition
`Downlink Domain Information`	Shows all the downlink entries: ■ Domain - Name of the downlink domain. ■ Domain Controller - Controller of the downlink domain. ■ Domain Controller MAC - MAC address of the domain controller. ■ Number of Links - Total number of links to this domain from the Node. ■ Number of Links Up - Total number of links that are in UP state. ■ Number of Links on This Node - Number of links terminating on this node. ■ Links are Blocked - 0 links are not blocked to the domain. 1 All links are blocked to the domain.
`Node Transaction List`	List of transactions from this downlink domain node. ■ Node - 0 links are not blocked to the domain. 1 All links are blocked to the domain. ■ Transaction ID - Transaction id for this node. ■ Domain List: Shows list of nodes in the current domain and their links to the downlink domain.: ■ Domain - Domain name of the downlink node. ■ Node - Name of the node in the current domain. ■ Ifindex - Interface index for the link from the node to the downlink domain. ■ Transaction ID - Transaction id of the node in the current domain. ■ Flags - As mentioned above.
`Up/Downlink Ports Information`	Shows all the configured up and down link ports on this node: ■ Port - Name of the local port. ■ Ifindex - Interface index of the local port. ■ VR ID - Virtual router id for the local port. ■ Port Status - Shows status of the local port on the Node as UP/DOWN. ■ Port State - AMF state of the local port. ■ Adjacent Node - nodename of the adjacent node. ■ Adjacent Internal ID - Unique node identifier of the remote node. ■ Adjacent Ifindex - Interface index for the port of adjacent AMF node. ■ Adjacent Board ID - Product identifier for the adjacent node. ■ Adjacent VR ID - Virtual router id for the port on adjacent AMF node. ■ Adjacent MAC - MAC address for the port on adjacent AMF node. ■ Adjacent Domain Controller - nodename of the Domain controller for Adjacent AMF node. ■ Adjacent Domain Controller MAC - MAC address of the Domain controller for Adjacent AMF node. ■ Port Forwarding State - Local port forwarding state Forwarding or Blocking. ■ Port BPDU Receive Count - count of AMF protocol PDU's received. ■ Port Sequence Number - hello sequence number, incremented every time the data in the hello packet changes. ■ Port Adjacent Sequence Number - remote ends sequence number used to check if we need to process this packet or just note it arrived. ■ Port Last Message Response - response from the remote neighbor to our last hello packet.

Note

You can manage your show output, or make it a more selective, by using a command modifier. For information on using show-command modifiers see **"Controlling "show" Command Output" on page 1.37**.

Related Commands

no debug all
clear atmf links statistics
show atmf

show atmf links statistics

This command displays details of the AMF links configured on the device and also displays statistics about the AMF packet exchanges between the devices.

It is also possible to display the AMF link configuration and packet exchange statistics for a specified interface.

This command can only be run on AMF master nodes

Syntax `show atmf links statistics [interface [<port_number>]]`

Parameter	Description
`interface`	Specifies that the command applies to a specific interface (port) or range of ports. Where both the interface and port number are unspecified, full statistics (not just those relating to ports will be displayed.
<port_number>	Enter the port number for which statistics are required. A port range or a static channel can also be specified. Where no port number is specified, statistics will be displayed for all ports on the switch.

Mode User Exec

Example 1 To display AMF link statistics for the whole switch, use the command:

```
switch1# show atmf links statistics
```

Figure 86-16: Sample output from the show atmf links statistics command

```
switch1# show atmf links statistics

ATMF Statistics:

                                    Receive          Transmit
-----------------------------------------------------------------
Crosslink Hello                       7                14
Crosslink Hello Domain                18               38
Crosslink Hello Uplink                3                12
Hello Link                            32               31
Hello Neighbor                        55               57
Hello Stack                           0                0
Database Description                  12               112
Database Request                      5                4
Database Reply                        0                5
Database Update                       35               9
Database Update Bitmap                0                10
Database Acknowlegde                  112              74
Transmit Fails                        0                0
Discards                              0                0
Total AMF Packets                     300              366

ATMF Database Statistics:

Database Entries                      18
Database Full Ages                    0

ATMF Packet Discards:

Type0       0               Type1       0               Type2       0
Type3       0               Type4       0               Type5       0
Type6       0               Type7       0               Type8       0
Type9       0               Type10      0               Type11      0
Type12      0               Type13      0               Type14      0
Type15      0               Type16      0               Type17      0
Type18      0               Type19      0               Type20      0
Type21      0               Type22      0

ATMF Virtual Link Statistics
Virtual                     Receive                       Transmit
Link          Receive       Dropped       Transmit        Dropped
-----------------------------------------------------------------
vlink1            0             0             0              0
vlink2        97383             0         36260              0
vlink6            0          3991             0              0
vlink16           0
```

Example 2 To display the AMF links statistics on interface port1.0.5, use the command:

```
switch1# show atmf links statistics interface
         port1.0.5
```

Figure 86-17: Sample output from the show atmf links statistics command for interface 1.0.5

```
switch1# show atmf links statistics interface port1.0.5

ATMF Port Statistics:

Transmit                                        Receive

port1.0.5  Crosslink Hello                      231             232
port1.0.5  Crosslink Hello Domain               116             116
port1.0.5  Crosslink Hello Uplink               116             115
port1.0.5  Hello Link                           0               0
```

Table 86-7: Parameter definitions from the show atmf links statistics command output

Parameter	Definition
`Receive`	Shows a count of AMF protocol packets received per message type.
`Transmit`	Shows the number of AMF protocol packets transmitted per message type.
`Database Entries`	Shows the number of AMF elements existing in the distributed database.
`Database Full Ages`	Shows the number of times the entries aged in the database.
`ATMF Packet Discards`	Shows the number of discarded packets of each type: ■ Type0: The number of discarded crosslink hello msgs received on a non crosslink port. ■ Type1: The number of discarded tx update packets - bad checksum. ■ Type2: The number of discarded tx update bitmap packets - bad checksum. ■ Type3: The number of discarded tx update packets - neighbor not in the correct state. ■ Type4: The number of discarded update packets - bad checksum. ■ Type5: The number of discarded update packets - neighbor not in the correct state. ■ Type6: The number of discarded update bitmap packets - bad checksum. ■ Type7: The number of discarded crosslink hello msgs received on a non crosslink port. ■ Type8: The number of discarded crosslink hello msg received on a port that is not in the correct state. ■ Type9: The number of discarded crosslink domain hello msgs received on a non crosslink port. ■ Type10: The number of discarded crosslink domain hello msgs received on a port that is not in the correct state. ■ Type11: The number of crosslink uplink hello msgs received on a non crosslink port. ■ Type12: The number of discarded crosslink uplink hello msgs ignored on a port that is not in the correct state. ■ Type13: The number of messages with an incorrect name for this AMF network. ■ Type14: The number of over-long packets received on a port. ■ Type15: The number of messages with a bad protocol version received on a port. ■ Type16: The number of messages with a bad packet checksum calculation received on a port. ■ Type17: The number of messages with a bad authentication type received on a port. ■ Type18: The number of messages with a bad simple password received on a port. ■ Type19: The number of discarded packets with an unsupported authentication type received on a port. ■ Type20: The number of discarded packets with an unknown neighbor received on a port.

Note

You can manage your show output, or make it a more selective, by using a command modifier. For information on using show-command modifiers see **"Controlling "show" Command Output" on page 1.37**.

Related Commands

no debug all
clear atmf links statistics
show atmf

show atmf memory

This command displays a summary of the AMF memory usage. It can only be run on AMF master nodes.

Syntax `show atmf memory`

Mode User Exec

Example To display AMF memory allocations on Node_1, use the command:

```
node_1# show atmf memory
```

Figure 86-18: Sample output from the show atmf memory command

```
node_1#show atmf memory

ATMF Memory Allocation:

Total memory allocated   : 30020 (bytes)
Total memory allocations : 77
 Line   1238  number      1  memory         28 (bytes)
 Line    244  number      2  memory         88 (bytes)
 Line   3753  number      2  memory       1872 (bytes)
 Line   1616  number      8  memory        320 (bytes)
 Line   1391  number      1  memory         60 (bytes)
 Line   1837  number     15  memory        600 (bytes)
 Line    288  number      1  memory      17716 (bytes)
 Line   3916  number      1  memory       1520 (bytes)
 Line   1623  number      8  memory        320 (bytes)
 Line   4477  number      1  memory       1520 (bytes)
 Line    659  number      2  memory        512 (bytes)
 Line   1844  number      6  memory        600 (bytes)
 Line   1749  number      1  memory         32 (bytes)
 Line    203  number      6  memory        600 (bytes)
 Line   4205  number      1  memory       1520 (bytes)
 Line    206  number      4  memory       1524 (bytes)
 Line    549  number      1  memory        232 (bytes)
 Line   3495  number      1  memory         56 (bytes)
 Line   2628  number      2  memory         72 (bytes)
 Line    678  number      1  memory         32 (bytes)
 Line   1423  number      1  memory         48 (bytes)
 Line   1733  number      3  memory        492 (bytes)
 Line   1611  number      8  memory        256 (bytes)
```

Figure 86-18: Sample output from the show atmf memory command (cont.)

```
ATMF Memory Deallocation:

Total memory deallocated   : 4958 (bytes)
Total memory deallocations : 45
 Line   1395  number      4  memory        400 (bytes)
 Line   1956  number      1  memory        164 (bytes)
 Line   1247  number      1  memory         52 (bytes)
 Line    876  number      2  memory         80 (bytes)
 Line    166  number      1  memory        232 (bytes)
 Line    415  number      7  memory        587 (bytes)
 Line    418  number      3  memory        300 (bytes)
 Line    822  number      2  memory         80 (bytes)
 Line   2341  number      4  memory        160 (bytes)
 Line   3025  number      2  memory         88 (bytes)
 Line    144  number      3  memory       1596 (bytes)
 Line    146  number      6  memory        312 (bytes)
 Line   2349  number      4  memory        160 (bytes)
 Line   1111  number      1  memory         59 (bytes)
 Line   1393  number      4  memory        688 (bytes)


-----------------------------------------------------
Total memory in use        : 4958 (bytes)
Total memory items         : 45
```

show atmf nodes

This command displays all nodes currently configured within the AMF network. It displays a topographical representation of the network infrastructure.

This command displays a summary of all virtual links currently in the running configuration.

Syntax `show atmf nodes`

Mode Privileged Exec

Example To display AMF information for all nodes in the AMF, use the command:

```
node_1# show atmf nodes
```

Figure 86-19: Sample output from the show atmf nodes command.

```
node1#show atmf nodes

Node Information:

  * = Local device

 SC = Switch Configuration:
  C = Chassis    S = Stackable    N = Standalone

  Node                Device             ATMF                              Node
  Name                Type               Master  SC  Parent                Depth
-------------------------------------------------------------------------------
  Building_1          AT-SBx8112           Y     C   -                      0
* Building_2          x900-12XT/S          Y     N   -                      0
  Bld1_Floor_1        SwitchBlade x908     N     S   Building_1             1
  Bld1_Floor_2        x600-24Ts/XP         N     N   Building_1             1
  Bld2_Floor_1        x610-24Ts-POE+       N     N   Building_1             1
  SW_Team1            x210-24GT            N     N   Bld1_Floor_2           2

Current ATMF node count 6
```

show atmf provision nodes

This command displays information about each provisioned node with details about date and time of creation, boot and configuration files available in the backup, and license files present in the provisioned backup. This includes nodes that have joined the network but are yet to run their first backup.

This command can only be run on AMF master nodes.

Syntax `show atmf provision nodes`

Mode Privileged Exec

Usage This command is only available on master nodes in the AMF network. The command will only work if provisioned nodes have already been set up. Otherwise, an error message is shown when the command is run.

Example To show the details of all the provisioned nodes in the backup use the command:

```
NodeName# show atmf provision nodes
```

Figure 86-20: Sample output from the show atmf provision nodes command

```
switch1#show atmf provision nodes

ATMF Provisioned Node Information:

Backup Media .............: SD (Total 3827.0MB, Free 3481.1MB)

Node Name                 : switch2
Date & Time               : 06-May-2014 & 23:25:44
Provision Path            : card:/atmf/provision_nodes

Boot configuration :
Current boot image        : x510-1766_atmf_backup.rel (file exists)
Backup  boot image        : x510-main-20140113-2.rel (file exists)
Default boot config       : flash:/default.cfg (file exists)
Current boot config       : flash:/abc.cfg (file exists)
Backup  boot config       : flash:/xyz.cfg (file exists)

Software Licenses :
Repository file           : ./.configs/.sw_v2.lic
                          : ./.configs/.swfeature.lic
Certificate file          : card:/atmf/nodes/awplus1/flash/.atmf-lic-cert
```

Related commands
- atmf provision node create
- atmf provision node clone
- atmf provision node configure boot config
- atmf provision node configure boot system
- show atmf backup

show atmf tech

This command collects and displays all the AMF command output. The command can thus be used to display a complete picture of an AMF network.

Syntax `show atmf tech`

Mode Privileged Exec

Example To display output for all AMF commands, use the command:

```
NodeName# show atmf tech
```

Figure 86-21: Sample output from the show atmf tech command.

```
node1#show atmf tech
ATMF Summary Information:

ATMF Status            : Enabled
Network Name           : ATMF_NET
Node Name              : node1
Role                   : Master
Current ATMF Nodes     : 8

ATMF Technical information:

Network Name                          : ATMF_NET
Domain                                : node1's domain
Node Depth                            : 0
Domain Flags                          : 0
Authentication Type                   : 0
MAC Address                           : 0014.2299.137d
Board ID                              : 287
Domain State                          : DomainController
Domain Controller                     : node1
Backup Domain Controller              : node2
Domain controller MAC                 : 0014.2299.137d
Parent Domain                         : -
Parent Domain Controller              : -
Parent Domain Controller MAC          : 0000.0000.0000
Number of Domain Events               : 0
Crosslink Ports Blocking              : 0
Uplink Ports Waiting on Sync          : 0
Crosslink Sequence Number             : 7
Domains Sequence Number               : 28
Uplink Sequence Number                : 2
Number of Crosslink Ports             : 1
Number of Domain Nodes                : 2
Number of Neighbors                   : 5
Number of Non Broadcast Neighbors     : 3
Number of Link State Entries          : 1
Number of Up Uplinks                  : 0
Number of Up Uplinks on This Node     : 0
DBE Checksum                          : 84fc6
Number of DBE Entries                 : 0
Management Domain Ifindex             : 4391
Management Domain VLAN                : 4091
Management ifindex                    : 4392
Management VLAN                       : 4092
...
...
```

Table 86-8: Parameter definitions from the show atmf tech command

Parameter	Definition
ATMF Status	Shows status of AMF feature on the Node as Enabled/Disabled.
Network Name	The name of the AMF network to which this node belongs.
Node Name	The name assigned to the node within the AMF network.
Role	The role configured on the switch within the AMF - either master or member.
Current ATMF Nodes	A count of the AMF nodes in the AMF network.
Node Address	The identity of a node (in the format name.atmf) that enables its access it from a remote location.
Node ID	A unique identifier assigned to an AMF node.
Node Depth	The number of nodes in path from this node to the core domain.
Domain State	A node's state within an AMF Domain - either controller or backup.
Recovery State	The AMF node recovery status. Indicates whether a node recovery is in progress on this device - either Auto, Manual, or None.
Management VLAN	The VLAN created for traffic between nodes of different domains (up/down links). VLAN ID - In this example VLAN 4092 is configured as the Management VLAN. Management Subnet - the Network prefix for the subnet. Management IP Address - the IP address allocated for this traffic. Management Mask - the Netmask used to create a subnet for this traffic 255.255.128.0 (= prefix /17)
Domain VLAN	The VLAN assigned for traffic between Nodes of same domain (crosslink). VLAN ID - In this example VLAN 4091 is configured as the domain VLAN. Domain Subnet - the Subnet address used for this traffic. Domain IP Address - the IP address allocated for this traffic. Domain Mask - the Netmask used to create a subnet for this traffic 255.255.128.0 (= prefix /17)
Device Type	Shows the Product Series Name.
ATMF Master	Indicates the nodes membership of the core domain (membership is indicated by Y)
SC	Shows switch configuration: ■ C - Chassis (such as SBx8100 series) ■ S - Stackable (VCS) ■ N - Standalone
Parent	A node to which connects to the present node's uplink, i.e. one layer higher in the hierarchy.
Node Depth	Shows the number of nodes in path from the current node to the Core domain.

Note The **show atmf tech** command can produce very large output. For this reason only the most significant terms are defined in this table.

show atmf working-set

This command displays the nodes that form the current AMF working-set.

Syntax `show atmf working-set`

Mode Privileged Exec

Example To show current members of the working-set, use the command:

```
ATMF_NETWORK[6]# show atmf working-set
```

Figure 86-22: Sample output from the show atmf working-set command.

```
ATMF Working Set Nodes:
node1, node2, node3, node4, node5, node6
Working set contains 6 nodes
```

Related Commands atmf working-set
show atmf
show atmf group

show debugging atmf

This command shows the debugging modes status for AMF.

Syntax `show debugging atmf`

Mode User Exec and Global Configuration

Example To display the AMF debugging status, use the command:

```
node_1# show debugging atmf
```

Figure 86-23: Sample output from the show debugging atmf command.

```
node1# show debugging atmf
ATMF debugging status:
ATMF link debugging is on
ATMF crosslink debugging is on
ATMF database debugging is on
ATMF neighbor debugging is on
ATMF packet debugging is on
ATMF error debugging is on
```

Related Commands debug atmf packet

show debugging atmf packet

This command shows details of AMF Packet debug command.

Syntax `show debugging atmf packet`

Mode User Exec and Global Configuration

Example To display the AMF packet debugging status, use the command:

```
node_1# show debug atmf packet
```

Figure 86-24: Sample output from the show debugging atmf packet command.

```
ATMF packet debugging is on
=== ATMF Packet Debugging Parameters===
Node Name: x900
Port name: port1.0.1
Limit: 500 packets
Direction: TX
Info Level: Level 2
Packet Type Bitmap:
2. Crosslink Hello BPDU pkt with downlink domain info
3. Crosslink Hello BPDU pkt with uplink info
4. Down and up link Hello BPDU pkts
6. Stack hello unicast pkts
8. DBE request
9. DBE update
10. DBE bitmap update
```

Related Commands **debug atmf**
debug atmf packet

show running-config atmf

This command displays the running system information that is specific to AMF.

Syntax `show running-config atmf`

Mode User Exec and Global Configuration

Example To display the current configuration of AMF, use the following commands:

```
node_1# show running-config atmf
```

Note You can manage your show output, or make it a more selective, by using a command modifier. For information on using show-command modifiers see **"Controlling "show" Command Output" on page 1.37**.

Related Commands **show running-config**
no debug all

switchport atmf-crosslink

This command configures the selected port or (statically) aggregated link to be an AMF crosslink. Running this command will automatically place the port or static aggregator into trunk mode (i.e. switchport mode trunk).

The connection between two AMF masters must utilize a crosslink. Crosslinks are used to carry the AMF control information between master nodes. Multiple crosslinks can be configured between two master nodes, but only one crosslink can be active at any particular time. All other crosslinks between masters will be placed in the blocking state, in order to prevent broadcast storms.

Use the **no** variant of this command to remove any crosslink that may exist for the selected port or aggregated link.

Syntax

```
switchport atmf-crosslink
no switchport atmf-crosslink
```

Mode Interface Configuration

Usage Crosslinks can be used anywhere within an AMF network. They have the effect of separating the AMF network into separate domains.

Where this command is used, it is also good practice to use the **switchport trunk native vlan** command with its parameter "**none**" selected. This is to prevent a network storm on a topology of ring connected switches.

Example 1 To make a switchport 1.0.1 an AMF crosslink, use the following commands:

```
Node_1# configure terminal
Node_1(config)# interface port1.0.1
Node_1(config-if)# switchport atmf-crosslink
```

Example 2 This example is shown twice. Example 2A is the most basic command sequence. Example 2B is a good practice equivalent that avoids problems such as broadcast storms that can otherwise occur.

Example 2A To make static aggregator sa1 an AMF crosslink, use the following commands:

```
Node_1# configure terminal
Node_1(config)# interface sa1
Node_1(config-if)# switchport atmf-crosslink
```

Example 2B To make static aggregator sa1 an AMF crosslink, use the following commands for good practice:

```
Node_1# configure terminal
Node_1(config)# interface sa1
Node_1(config-if)# switchport atmf-crosslink
Node_1(config-if)# switchport trunk allowed vlan add 2
Node_1(config-if)# switchport trunk native vlan none
```

In this example VLAN 2 is assigned to the static aggregator, and the native VLAN (VLAN 1) is explicitly excluded from the aggregated ports and the crosslink assigned to it.

Note The AMF management and domain VLANs are automatically added to the aggregator and the crosslink.

Related Commands **show atmf links statistics**

switchport atmf-link

This command enables you to configure a port or aggregator to be an AMF uplink/downlink. Running this command will automatically place the port or aggregator into trunk mode.

Use the **no** variant of this command to remove any AMF-link that may exist for the selected port or aggregated link.

Syntax

```
switchport atmf-link
no switchport atmf-link
```

Mode Interface Configuration

Example To make a switchport 1.0.1 an AMF crosslink, use the following commands

```
Node_1# configure terminal
Node_1(config)# interface port1.0.1
Node_1(config-if)# switchport atmf-link
```

type atmf node

This command configures a trigger to be activated at an AMF node join event or leave event.

Syntax `type atmf node {join|leave}`

Parameter	Description
`join`	AMF node join event.
`leave`	AMF node leave event.

Mode Trigger Configuration

Caution **Only configure this trigger on one device because it is a network wide event.**

Example 1 To configure trigger 5 to activate at an AMF node leave event, use the following commands. In this example the command is entered on node-1:

```
node1(config)# trigger 5
node1(config-trigger) type atmf node leave
```

Example 2 The following commands will configure trigger 5 to activate if an AMF node join event occurs on any node within the working set:

```
node1# atmf working-set group all
```

This command returns the following display:

```
====================
node1, node2, node3:
====================

Working set join
```

Note that the running the above command changes the prompt from the name of the local node, to the name of the AMF-Network followed, in square brackets, by the number of member nodes in the working set.

```
AMF-Net[3]# conf t
AMF-Net[3](config)# trigger 5
AMF-Net[3](config-trigger)# type atmf node leave
AMF-Net[3](config-trigger)# description "E-mail on AMF Exit"
AMF-Net[3](config-trigger)# active
```

Enter the name of the script to run at the trigger event.

```
AMF-Net[3](config-trigger)# script 1 email_me.scp
AMF-Net[3](config-trigger)# end
```

Display the trigger configurations

```
AMF-Net[3]# show trigger
```

This command returns the following display:

```
=======
node1:
=======

TR# Type & Details     Description          Ac Te Tr Repeat     #Scr Days/Date
-------------------------------------------------------------------------------
001 Periodic (2 min)   Periodic Status Chk  Y  N  Y Continuous  1   smtwtfs
005 ATMF node (leave)  E-mail on ATMF Exit  Y  N  Y Continuous  1   smtwtfs
-------------------------------------------------------------------------------


==============
Node2, Node3,
==============

TR# Type & Details     Description          Ac Te Tr Repeat     #Scr Days/Date
-------------------------------------------------------------------------------
005 ATMF node (leave)  E-mail on ATMF Exit  Y  N  Y Continuous  1   smtwtfs
-------------------------------------------------------------------------------
```

Display the triggers configured on each of the nodes in the AMF Network.

```
AMF-Net[3]# show running-config trigger
```

This command returns the following display:

```
========
Node1:
========

trigger 1
 type periodic 2
 script 1 atmf.scp
trigger 5
 type atmf node leave
description "E-mail on ATMF Exit"
 script 1 email_me.scp
!


============
Node2, Node3:
============

trigger 5
 type atmf node leave
description "E-mail on ATMF Exit"
 script 1 email_me.scp
!
```

Related Commands **show trigger**

undebug atmf

This command is an alias for the **no variant of the debug atmf** command on page 86.52.

The IPv4 addresses shown may include those specified for documentation purposes in RFC 5737: 192.0.2.0/24, 198.51.100.0/24, 203.0.113.0/24. These addresses should not be used for practical networks (other than for testing purposes), nor should they appear in any public network.

Chapter 87: NTP Introduction and Configuration

Introduction 87.2
Overview 87.2
NTP on the Switch 87.3
Troubleshooting 87.4
Configuration Example 87.4

Introduction

This chapter describes the Network Time Protocol (NTP) service provided by the switch, and how to configure and monitor NTP on the switch.

NTP is a protocol for synchronizing the time clocks on a collection of network devices using a distributed client/server mechanism. NTP uses UDP (User Datagram Protocol) as the transport mechanism. NTP evolved from the Time Protocol (RFC 868) and the ICMP Timestamp message (RFC 792).

NTP provides protocol mechanisms to specify the precision and estimated error of the local clock and the characteristics of the reference clock to which it may be synchronized.

For detailed information about the commands used to configure NTP, see **Chapter 88, NTP Commands**.

Overview

NTP uses a subnetwork with primary reference clocks, gateways, secondary reference clocks, and local hosts. These are organized into a hierarchy with the more accurate clocks near the top and less accurate ones near the bottom.

A number of primary reference clocks, synchronized to national standards, are connected to widely accessible resources (such as backbone gateways or switches) operating as primary time servers. The primary time servers use NTP between them to crosscheck clocks, to mitigate errors due to equipment or propagation failures, and to distribute time information to local secondary time servers. The secondary time servers redistribute the time information to the remaining local hosts.

The hierarchical organization and distribution of time information reduces the protocol overhead, and allows selected hosts to be equipped with cheaper but less accurate clocks. NTP provides information which organizes this hierarchy on the basis of precision or estimated error.

- An NTP entity may be in one of the following operating modes; however, the switch's implementation of NTP supports two modes: client and server.
- An NTP entity operating in a client mode sends periodic messages to its peers, requesting synchronization by its peers.
- An NTP entity enters the server mode temporarily when it receives a client request message from one of its peers, and remains in server mode until the reply to the request has been transmitted.
- An NTP entity operating in symmetric active mode sends messages announcing its willingness to synchronize and be synchronized by its peers.
- An NTP entity enters symmetric passive mode in response to a message from a peer operating in Symmetric Active mode. An NTP entity operating in this mode announces its willingness to synchronize and be synchronized by its peers.
- An NTP entity operating in broadcast mode periodically sends messages announcing its willingness to synchronize all of its peers but not to be synchronized by any of them.

The same message format is used for both requests and replies. When a request is received, the server interchanges addresses and ports, fills in or overwrites certain fields in the message, recalculates the checksum, and returns it immediately. The information included in the NTP message allows each client/ server peer to determine the timekeeping characteristics of its peers, including the expected accuracies of their clocks. Each peer uses this information and selects the best time from possibly several other clocks, updates the local clock, and estimates its accuracy.

There is no provision in NTP for peer discovery, acquisition, or authentication. Data integrity is provided by the IP and UDP checksums. No reachability, circuit-management, duplicate-detection, or retransmission facilities are provided or necessary.

By its very nature clock synchronization requires long periods of time (hours or days) and multiple comparisons in order to maintain accurate timekeeping. The more comparisons performed, the greater the accuracy of the timekeeping.

NTP on the Switch

The implementation of NTP on the switch is based on the following RFCs:

- RFC 958, Network Time Protocol (NTP)
- RFC 1305, Network Time Protocol (Version 3) Specification, Implementation and Analysis
- RFC 1510, The Kerberos Network Authentication Service (V5)

Two modes of operation are supported: client and server. The switch is in client mode most of the time where it polls the configured peer at least once every preconfigured minimum time period.

The peer that the switch refers to must be a more accurate clock source than the switch itself or another switch directly connected to a more accurate clock source. The switch operates as a secondary time server. It cannot operate as a primary time server unless the primary clock source is operating in server mode. A primary clock source usually operates in broadcast mode, which is not supported by the switch's implementation of NTP. When the switch receives a valid reply from the peer, it synchronizes its own internal clock according to the information from the reply.

If the switch receives a synchronization request from an NTP client, it temporarily changes to server mode. It replies to the request with the current time from the switch's internal clock along with other information useful for synchronization. The switch's internal clock is accurate to 0.005 seconds.

Troubleshooting

Problem The switch is not assigning the time to devices on the LAN.

Solutions

- Check that the NTP peer's IP address is entered correctly.
- Check that the NTP peer can reach the switch, by pinging the switch from the NTP peer.

Problem The switch's clock does not synchronize with the NTP peer.

Solution

- The switch's clock can synchronize with the NTP peer only when its initial time is similar to the NTP peer's time (after setting the UTC offset). Manually set the switch's time so that it is approximately correct, and enable NTP again.
- Check that the UTC offset is correct.

Problem The switch's time is incorrect, even though it assigns the correct time to devices on the LAN.

Solution The UTC offset is probably incorrect, or needs to be adjusted for the beginning or end of summer time.

Configuration Example

NTP requires the IP module to be enabled and configured correctly.

The switch's implementation of NTP supports two modes: client and server mode. When a synchronization request is received from a client (e.g. a PC on a LAN), the switch enters server mode and responds with time information derived from the switch's own internal clock. Periodically the switch enters client mode, sending synchronization requests to a predefined peer to synchronize its own internal clock. The peer is assumed to be a primary clock source or another switch connected directly to a primary clock source.

This example illustrates how to configure two switches, one at a Head Office and one at a Regional Office, to provide a network time service. The Head Office switch is connected to a primary time server and provides the most accurate time information. The switch at the Regional Office uses the Head Office switch as its peer to avoid the cost of an additional WAN connection but provides slightly less accurate time information.

To configure NTP on the switch, the NTP module must be enabled and an NTP peer must be defined. NTP transfers time information in UTC format.

To set the switch to automatically change the time when summer time starts and ends, enable a summer time offset setting.

Example configuration parameters for a network time service:

Site	Regional Office	Head Office
Switch name	RG1	HO1
IP Address of Switch	10.5.35.114	10.12.25.4
IP Address of Peer	10.5.35.113	172.16.7.3

Step 1: Enable NTP and define the NTP peer.

The NTP feature must be enabled on all switches that are to provide a network time service. Each switch must have a peer defined where the switch synchronizes its own internal clock. Enable NTP on the Head Office switch and specify a primary time server as the peer by using the commands:

```
awplus# configure terminal
awplus(config)# ntp peer 172.16.7.3
```

Note that you can also specify an IPv6 address for an NTP peer:

```
awplus# configure terminal
awplus(config)# ntp peer 2001:0db8:010d::1
```

Step 2: Configure the NTP parameters.

On each switch, the offset of local time from UTC time must be specified. In this example, both switches are in the same time zone, which is 12 hours ahead of UTC time. Use the following commands on both switches:

```
awplus(config)# clock timezone utc plus 12
```

Note that the range of offset is <0-12>.

Step 3: Check the NTP configuration.

Check the NTP configuration on each switch by using the command:

```
awplus# show ntp status
```

This command displays the following information on the Head Office switch.

```
Clock is synchronized, stratum 0, actual frequency is 0.0000
Hz, precision is 20 reference time is 00000000.00000000
(6:28:16.000 UTC Fri Feb  7 2036)clock offset is 0.000 msec,
root delay is 0.000 msec root dispersion is 0.000 msec,
```

Chapter 88: NTP Commands

Command List 88.2
ntp access-group 88.2
ntp authenticate 88.3
ntp authentication-key 88.4
ntp broadcastdelay 88.5
ntp master 88.6
ntp peer 88.7
ntp server 88.9
ntp source 88.11
ntp trusted-key 88.13
show counter ntp 88.14
show ntp associations 88.15
show ntp status 88.16

Command List

This chapter provides an alphabetical reference for commands used to configure the Network Time Protocol (NTP). For more information, see **Chapter 87, NTP Introduction and Configuration**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

ntp access-group

This command creates an NTP access group, and applies a basic IP access list to it. This allows you to control access to NTP services.

The **no** variant of this command removes the configured NTP access group.

Syntax

```
ntp access-group [peer|query-only|serve|serve-only] [<1-99>|
    <1300-1999>]

no ntp access-group [peer|query-only|serve|serve-only]
```

Parameter	Description
`peer`	Allows time requests and NTP control queries, and allows the system to synchronize itself to a system whose address passes the access list criteria.
`query-only`	Allows only NTP control queries from a system whose address passes the access list criteria.
`serve`	Allows time requests and NTP control queries, but does not allow the system to synchronize itself to a system whose address passes the access list criteria.
`serve-only`	Allows only time requests from a system whose address passes the access list criteria.
<1-99>	Standard IP access list.
<1300-1999>	Expanded IP access list.

Mode Global Configuration

Examples To create an NTP peer access group for an extended IP access list, use the commands:

```
awplus# configure terminal
awplus(config)# ntp access-group peer 1998
```

To disable the NTP peer access group created above, use the commands:

```
awplus# configure terminal
awplus(config)# no ntp access-group peer
```

ntp authenticate

This command enables NTP authentication. This allows NTP to authenticate the associations with other systems for security purposes.

The **no** variant of this command disables NTP authentication.

Syntax

```
ntp authenticate
no ntp authenticate
```

Mode Global Configuration

Examples To enable NTP authentication, use the commands:

```
awplus# configure terminal
awplus(config)# ntp authenticate
```

To disable NTP authentication, use the commands:

```
awplus# configure terminal
awplus(config)# no ntp authenticate
```

ntp authentication-key

This command defines each of the authentication keys. Each key has a key number, a type, and a value. Currently, the only key type supported is MD5.

The **no** variant of this disables the authentication key assigned previously using **ntp authentication-key**.

Syntax

```
ntp authentication-key <keynumber> md5 <key>
no ntp authentication-key <keynumber> md5 <key>
```

Parameter	Description
<keynumber>	*<1-4294967295>* The key number.
<key>	The authentication key.

Mode Global Configuration

Examples To define an authentication key number `134343` and a key value `mystring`, use the commands:

```
awplus# configure terminal
awplus(config)# ntp authentication-key 134343 md5 mystring
```

To disable the authentication key number `134343` with the key value `mystring`, use the commands:

```
awplus# configure terminal
awplus(config)# no ntp authentication-key 134343 md5 mystring
```

ntp broadcastdelay

Use this command to set the estimated round-trip delay for broadcast packets.

Use the **no** variant of this command to reset the round-trip delay for broadcast packets to the default offset of 0 microseconds.

Syntax

```
ntp broadcastdelay <delay>
no ntp broadcastdelay
```

Parameter	Description
<delay>	*<1-999999>* The broadcast delay in microseconds.

Default 0 microsecond offset, which can only be applied with the **no** variant of this command.

Mode Global Configuration

Examples To set the estimated round-trip delay to 23464 microseconds for broadcast packets, use these commands:

```
awplus# configure terminal
awplus(config)# ntp broadcastdelay 23464
```

To reset the estimated round-trip delay for broadcast packets to the default setting (0 microseconds), use these commands:

```
awplus# configure terminal
awplus(config)# no ntp broadcastdelay
```

ntp master

Use this command to make the device to be an authoritative NTP server, even if the system is not synchronized to an outside time source. Note that no stratum number is set by default.

Use the **no** variant of this command to stop the device being the designated NTP server.

Syntax

```
ntp master [<stratum>]
no ntp master
```

Parameter	Description
<stratum>	*<1-15>* The stratum number defines the configured level that is set for this master within the NTP hierarchy.

Mode Global Configuration

Usage The stratum number is null by default and must be set using this command. The stratum levels define the distance from the reference clock and exist to prevent cycles in the hierarchy. Stratum 1 is used to indicate time servers, which are more accurate than Stratum 2 servers. For more information on the Network Time Protocol go to: **www.ntp.org//**

Examples To stop the switch from being the designated NTP server use the commands:

```
awplus# configure terminal
awplus(config)# no ntp master
```

To make the switch the designated NTP server with stratum number 2 use the commands:

```
awplus# configure terminal
awplus(config)# ntp master 2
```

ntp peer

Use this command to configure an NTP peer association. An NTP association is a peer association if this system is willing to either synchronize to the other system, or allow the other system to synchronize to it.

Use the **no** variant of this command to remove the configured NTP peer association.

Syntax

```
ntp peer {<peeraddress>|<peername>}
ntp peer {<peeraddress>|<peername>} [prefer] [key <key>]
[version <version>]
no ntp peer {<peeraddress>|<peername>}
```

Parameter	Description
<peeraddress>	Specify the IP address of the peer, entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X:X` for an IPv6 address.
<peername>	Specify the peer hostname. The peer hostname can resolve to an IPv4 and an IPv6 address.
`prefer`	Prefer this peer when possible.
`key` *<key>*	*<1-4294967295>* Configure the peer authentication key.
`version` *<version>*	*<1-4>* Configure for this NTP version.

Mode Global Configuration

Examples See the following commands for options to configure NTP peer association, key and NTP version for the peer with an IPv4 address of `192.0.2.23`:

```
awplus# configure terminal
awplus(config)# ntp peer 192.0.2.23
awplus(config)# ntp peer 192.0.2.23 prefer
awplus(config)# ntp peer 192.0.2.23 prefer version 4
awplus(config)# ntp peer 192.0.2.23 prefer version 4 key 1234
awplus(config)# ntp peer 192.0.2.23 version 4 key 1234
awplus(config)# ntp peer 192.0.2.23 version 4
awplus(config)# ntp peer 192.0.2.23 key 1234
```

To remove an NTP peer association for this peer with an IPv4 address of `192.0.2.23`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ntp peer 192.0.2.23
```

See the following commands for options to configure NTP peer association, key and NTP version for the peer with an IPv6 address of `2001:0db8:010d::1`:

```
awplus# configure terminal
awplus(config)# ntp peer 2001:0db8:010d::1
awplus(config)# ntp peer 2001:0db8:010d::1 prefer
awplus(config)# ntp peer 2001:0db8:010d::1 prefer version 4
awplus(config)# ntp peer 2001:0db8:010d::1 prefer version 4 key
1234
awplus(config)# ntp peer 2001:0db8:010d::1 version 4 key 1234
awplus(config)# ntp peer 2001:0db8:010d::1 version 4
awplus(config)# ntp peer 2001:0db8:010d::1 key 1234
```

To remove an NTP peer association for this peer with an IPv6 address of `2001:0db8:010d::1`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ntp peer 2001:0db8:010d::1
```

Related Commands **ntp server**
ntp source

ntp server

Use this command to configure an NTP server. This means that this system will synchronize to the other system, and not vice versa.

Use the **no** variant of this command to remove the configured NTP server.

Syntax

```
ntp server {<serveraddress>|<servername>}
ntp server {<serveraddress>|<servername>} [prefer] [key <key>]
   [version <version>]
no ntp server {<serveraddress>|<servername>}
```

Parameter	Description
<serveraddress>	Specify the IP address of the peer, entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X.X` for an IPv6 address.
<servername>	Specify the server hostname. The server hostname can resolve to an IPv4 and an IPv6 address.
`prefer`	Prefer this server when possible.
`key` *<key>*	*<1-4294967295>* Configure the server authentication key.
`version` *<version>*	*<1-4>* Configure for this NTP version.

Mode Global Configuration

Examples See the following commands for options to configure an NTP server association, key and NTP version for the server with an IPv4 address of `192.0.1.23`:

```
awplus# configure terminal
awplus(config)# ntp server 192.0.1.23
awplus(config)# ntp server 192.0.1.23 prefer
awplus(config)# ntp server 192.0.1.23 prefer version 4
awplus(config)# ntp server 192.0.1.23 prefer version 4 key 1234
awplus(config)# ntp server 192.0.1.23 version 4 key 1234
awplus(config)# ntp server 192.0.1.23 version 4
awplus(config)# ntp server 192.0.1.23 key 1234
```

To remove an NTP peer association for this peer with an IPv4 address of `192.0.1.23`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ntp server 192.0.1.23
```

See the following commands for options to configure an NTP server association, key and NTP version for the server with an IPv6 address of `2001:0db8:010e::2`:

```
awplus# configure terminal
awplus(config)# ntp server 2001:0db8:010e::2
awplus(config)# ntp server 2001:0db8:010e::2 prefer
awplus(config)# ntp server 2001:0db8:010e::2 prefer version 4
awplus(config)# ntp server 2001:0db8:010e::2 prefer version 4
key 1234
awplus(config)# ntp server 2001:0db8:010e::2 version 4 key 1234
awplus(config)# ntp server 22001:0db8:010e::2 version 4
awplus(config)# ntp server 2001:0db8:010e::2 key 1234
```

To remove an NTP peer association for this peer with an IPv6 address of `2001:0db8:010e::2`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ntp server 2001:0db8:010e::2
```

Related Commands **ntp peer**
ntp source

ntp source

Use this command to configure an IPv4 or an IPv6 address for the NTP source interface. This command defines the socket used for NTP messages, and only applies to NTP client behavior.

Use the **no** variant of this command to remove the configured IPv4 or IPv6 address from the NTP source interface.

Syntax

```
ntp source <source-address>
no ntp source
```

Parameter	Description
<source-address>	Specify the IP address of the NTP source interface, entered in the form `A.B.C.D` for an IPv4 address, or in the form `X:X::X.X` for an IPv6 address.

Default An IP address is selected based on the most appropriate egress interface used to reach the NTP peer if a configured NTP client source IP address is unavailable or is an invalid IP address.

Mode Global Configuration

Usage Adding an IPv4 or an IPv6 address allows you to select which source interface NTP uses for peering. The IPv4 or IPv6 address configured using this command is matched to the interface.

When selecting a source IP address to use for NTP messages to the peer, if the configured NTP client source IP address is unavailable then default behavior will apply, and an alternative source IP address is automatically selected. This IP address is based on the most appropriate egress interface used to reach the NTP peer. The configured NTP client source IP may be unavailable if the interface is down, or an invalid IP address is configured that does not reside on the device.

Note that this command only applies to NTP client behavior. The egress interface that the NTP messages use to reach the NTP server determined by the **ntp peer** and **ntp server** commands.

Examples To configure the NTP source interface with the IPv4 address `192.0.1.23`, enter the commands:

```
awplus# configure terminal
awplus(config)# ntp source 192.0.1.23
```

To configure the NTP source interface with the IPv6 address `2001:0db8:010e::2`, enter the commands:

```
awplus# configure terminal
awplus(config)# ntp source 2001:0db8:010e::2
```

To remove a configured address for the NTP source interface, use the following commands:

```
awplus# configure terminal
awplus(config)# no ntp source
```

Related Commands **ntp peer**
ntp server

ntp trusted-key

This command defines a list of trusted authentication keys. If a key is trusted, this system will be ready to synchronize to a system that uses this key in its NTP packets.

Use the **no** variant of this command to remove a configured trusted authentication key.

Syntax

```
ntp trusted-key <1-4294967295>
no ntp trusted-key <1-4294967295>
```

Parameter	Description
<1-4294967295>	The specific key number.

Mode Global Configuration

Examples To define a trusted authentication key numbered `234675`, use the following commands:

```
awplus# configure terminal
awplus(config)# ntp trusted-key 234676
```

To remove the trusted authentication key numbered `234675`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ntp trusted-key 234676
```

show counter ntp

This command displays packet counters for NTP.

Syntax `show counter ntp`

Mode User Exec and Privileged Exec

Output **Figure 88-1: Example output from the show counter ntp command**

```
NTP counters
Pkts Sent               ......... 0
Pkts Received           ......... 70958
Pkts Processed          ......... 0
Pkts current version    ......... 0
Pkts old version        ......... 0
Pkts unknown version    ......... 0
Pkts access denied      ......... 70958
Pkts bad length         ......... 0
Pkts bad auth           ......... 0
Pkts rate exceed        ......... 0
```

Table 88-1: Parameters in the output from the show counter ntp command

Parameter	Description
`Pkts Sent`	Total number of NTP client and server packets sent by your device.
`Pkts Received`	Total number of NTP client and server packets received by your device.
`Pkts Processed`	The number of packets processed by NTP. NTP processes a packet once it has determined that the packet is valid by checking factors such as the packet's authentication, format, access rights and version.
`Pkts current version`	The number of version 4 NTP packets received.
`Pkts old version`	The number of NTP packets received that are from an older version, down to version 1, of NTP. NTP is compatible with these versions and processes these packets.
`Pkts unknown version`	The number of NTP packets received that are an earlier version than version 1, or a higher version than version 4. NTP cannot process these packets.
`Pkts access denied`	The number of NTP packets received that do not match any access list statements in the NTP access-groups. NTP drops these packets.
`Pkts bad length`	The number of NTP packets received that do not conform to the standard packet length. NTP drops these packets.
`Pkts bad auth`	The number of NTP packets received that failed authentication. NTP drops these packets. Packets can only fail authentication if NTP authentication is enabled with the **ntp authenticate** command.
`Pkts rate exceed`	The number of packets dropped because the packet rate exceeded its limits.

Example To display counters for NTP, use the command:

```
awplus# show counter ntp
```

show ntp associations

Use this command to display the status of NTP associations. Use the detail option for displaying detailed information about the associations.

Syntax `show ntp associations [detail]`

Mode User Exec and Privileged Exec

Example See the sample output of the **show ntp associations** and **show ntp associations detail** commands displaying the status of NTP associations.

Figure 88-2: Example output from the show ntp associations command

```
awplus#show ntp associations
  address          ref clock       st  when  poll reach   delay  offset    disp
 ~192.0.2.23       INIT            16     -   512   000     0.0     0.0     0.0
 * master (synced), # master (unsynced), + selected, - candidate, ~ configured
awplus#
```

Figure 88-3: Example output from the show ntp associations detail command

```
awplus#show ntp associations detail
192.0.2.23 configured, sane, valid, leap_sub, stratum 16
ref ID INIT, time 00000000.00000000 (06:28:16.000 UTC Thu Feb  7 2036)
our mode client, peer mode unspec, our poll intvl 512, peer poll intvl 1024
root delay 0.00 msec, root disp 0.00, reach 000,
delay 0.00 msec, offset 0.0000 msec, dispersion 0.00
precision 2**-19,
org time 00000000.00000000 (06:28:16.000 UTC Thu Feb  7 2036)
rcv time 00000000.00000000 (06:28:16.000 UTC Thu Feb  7 2036)
xmt time cf11f2a4.cedde5e4 (00:39:00.808 UTC Tue Feb  2 2010)
filtdelay =  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
filtoffset =  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
filterror =  16000.00  16000.00  16000.00  16000.00  16000.00  16000.00  16000.0
0  16000.00
```

Table 88-2: Parameters in the output from the show ntp associations command

Parameter	Description
`address`	Peer IP address
`ref clock`	IP address for reference clock
`st`	Stratum. The number of hops between the server and the accurate time source.
`poll`	Time between NTP requests from the device to the server.
`reach`	Shows whether or not the NTP server responded to the last request.
`delay`	Round trip delay between the device and the server.
`offset`	Difference between the device clock and the server clock.
`disp`	Lowest measure of error associated with peer offset based on delay.

show ntp status

Use this command to display the status of the Network Time Protocol (NTP).

Syntax `show ntp status`

Mode User Exec and Privileged Exec

Example See the sample output of the **show ntp status** command displaying information about the Network Time Protocol.

Figure 88-4: Example output from the show ntp status command

```
awplus#sh ntp status
Clock is synchronized, stratum 3, reference is 127.127.1.0
actual frequency is 0.0000 Hz, precision is 2**-19
reference time is cf11f3f2.c7c081a1 (00:44:34.780 UTC Tue Feb  2
2010)
clock offset is 0.000 msec, root delay is 0.000 msec
root dispersion is 7947729.000 msec,
awplus#
```

Chapter 89: Dynamic Host Configuration Protocol (DHCP) Introduction

Introduction 89.2
BOOTP 89.2
DHCP 89.2
DHCP Relay Agents 89.3
Configuring the DHCP Server 89.4
Create the Pool 89.4
Define the Network 89.4
Define the Range 89.5
Set the Lease 89.5
Enable DHCP Leasequery 89.5
Set the Options 89.7
DHCP Lease Probing 89.8
DHCP Relay Agent Introduction 89.9
Configuring the DHCP Relay Agent 89.9
DHCP Relay Agent Information Option (Option 82) 89.10
DHCPv6 Relay Agent Notification for DHCPv6 PD 89.13
Configuring the DHCP Client 89.14
Clearing Dynamically Allocated Lease Bindings 89.14

Introduction

This chapter describes the Dynamic Host Configuration Protocol (DHCP) support provided by your device. This includes how to configure your device to:

- act as a DHCP and BOOTP server
- act as a DHCP relay agent
- use the DHCP client to obtain IP addresses for its own interfaces

Note that you can configure your device to operate as both a DHCP relay agent and a DHCP/BOOTP server.

BOOTP

Bootstrap Protocol (BOOTP) is a UDP-based protocol that enables a booting host to dynamically configure itself without external interventions. A BOOTP server responds to requests from BOOTP clients for configuration information, such as the IP address the client should use. BOOTP is defined in RFC 951, Bootstrap Protocol (BOOTP).

RFC 1542, Clarifications and Extensions for the Bootstrap Protocol, defines extensions to the BOOTP protocol, including the behavior of a DHCP relay agent.

DHCP

DHCP is widely used to dynamically assign host IP addresses from a centralized server that reduces the overhead of administrating IP addresses. DHCP helps conserve the limited IP address space because IP addresses no longer need to be permanently assigned to hosts. DHCP centrally manages IP address assignment for a large number of subscribers.

DHCP is based on BOOTP, and is defined in RFC 2131. It extends the BOOTP mechanism by providing:

- a method for passing configuration information to hosts on a TCP/IP network
- automatic allocation of reusable network addresses
- other additional configuration options

When your device is configured as a DHCP server, it allocates IP addresses and other IP configuration parameters to clients (hosts), when the client requests them. This lets you configure your IP network without manually configuring every client. Note that each client must also be configured to receive its IP address automatically.

As well as addresses, a DHCP server assigns a wide range of parameters to clients, including subnet information and mask, domain and hostname, server addresses, keepalive times, MTUs, boot settings, encapsulation settings, time settings, and TCP settings.

DHCP is designed to interoperate with BOOTP clients and DHCP clients, without the BOOTP clients needing any change to their initialization software.

DHCP Relay Agents

DHCP relay agents pass BOOTP and DHCP messages between servers and clients. Networks where the DHCP or BOOTP server does not reside on the same IP subnet as its clients need the intermediate routers to act as relay agents. A maximum number of 400 DHCP relay agents (one per interface) can be configured on the device. Once this limit has been reached, any further attempts to configure DHCP relay agents will not be successful.

Configuring the DHCP Server

The DHCP server uses **address pools** when responding to DHCP client requests. Address pools contains specific IP configuration details that the DHCP server can allocate to a client. You can configure multiple address pools on the device for different networks.

Note that you cannot configure DHCP Server and DHCP Relay on the same device.

To configure a pool, you must:

- **Create the Pool** and enter its configuration mode.
- **Define the Network** the pool applies to.
- **Define the Range** of IP addresses that the server can allocate to clients. You can specify multiple address ranges for each pool.
- **Set the Lease** for the clients. This defines whether the clients receive a dynamic, permanent, or static IP address.
- **Set the Options** (standard and user-defined) that the clients of a pool require when configuring their IP details.

After configuring the address pools, you can then enable the DHCP server by using the command:

```
awplus(config)# service dhcp-server
```

For networks where you do not want the server to respond to BOOTP requests, you can configure the DHCP server so that it ignores them, by using the command:

```
awplus(config)# ip dhcp bootp ignore
```

Create the Pool

A DHCP pool is identified by a name. To create a DHCP pool and enter the DHCP Configuration mode for the pool, use the command:

```
awplus(config)# ip dhcp pool <pool-name>
awplus(dhcp-config)#
```

Define the Network

Define the network that the DHCP clients are in. You can define one network per address pool. Use the following command to define the network after defining the DHCP pool first:

```
awplus(dhcp-config)# network
```

- For remote clients, set the network address to the network of the remote clients. The **network** command does not need to match a specific interface's network, because the DHCP server listens on all IP interfaces for DHCP requests.
- For locally connected clients, ensure that the desired interface has an IP address and subnet mask defined; use the **ip address IPADDR** command to set a static address. Enter the configuration mode for the pool, and set the DHCP address pool's network to match the interface's network. Pools that span multiple interfaces are possible only if the interface networks are contiguous.

Define the Range

Configure an IP address range for the pool. This range must be in the same subnet as the pool's network setting. Use the command:

```
awplus(dhcp-config)# range <ip-address> [<ip-address>]
```

The first IPv4 address specifies the **low end of the range, while the second IP address is the high end.** You can set the range to a single IP address by specifying only one IP address.

Set the Lease

The DHCP server assigns IP settings to hosts for specific times (the lease time). Each DCHP pool has one lease time setting. You can use DHCP to allocate the following types of addresses:

- A **dynamic** IP addresses

  These are available to a host for a limited amount of time. When the lease expires, the server can reallocate the IP address to another device. To set the lease time for the DHCP pool so that it assigns dynamic IP addresses, use the command:

  ```
  awplus(dhcp-config)# lease <days> <hours> <minutes>
                       [<seconds>]
  ```

- A **permanent** IP addresses

  These are available to a host for an unlimited amount of time. To set the lease time to assign permanent IP addresses, use the command:

  ```
  awplus(dhcp-config)# lease infinite
  ```

- A **static** IP addresses

  These are allocated to a particular client. The DHCP server recognizes the client by its MAC address. This lets you use DHCP to manage most of your network automatically, while having unchanging IP addresses on key devices such as servers. To assign a static IP address to a device, use the command:

  ```
  awplus(dhcp-config)# host <ip-address> <mac-address>
  ```

BOOTP requests can be satisfied by pools with leases set to infinity.

Enable DHCP Leasequery

The DHCP Leasequery protocol (RFC 4388) allows a device or process, for example a DHCP relay agent, to obtain IP address information directly from the DHCP server using DHCPLEASEQUERY messages.

DHCPLEASEQUERY messages support three query regimes:

- IP address

  Only an IP address is supplied in the DHCPLEASEQUERY message. The DHCP server will return any information that it has on the most recent client to have been assigned that IP address.

- MAC address

Only a MAC address is supplied in the DHCPLEASEQUERY message. The DHCP server will return any information that it has on the IP address most recently accessed by a client with that MAC address. Also, the DHCP server may supply additional IP addresses that have been associated with that MAC address in different subnets.

- Client identifier option

  Only a Client identifier option is supplied in the DHCPLEASEQUERY message. The DHCP server will return any information that it has on the IP address most recently accessed by a client with that Client identifier. Also, the DHCP server may supply additional IP addresses that have been associated with Client identifier in different subnets.

An AlliedWare Plus DHCP server implementing DHCP Leasequery supports all three query regimes.

If the DHCP Leasequery feature is enabled, when a DHCP relay agent needs to know the location of an IP endpoint and sends a DHCPLEASEQUERY message, the DHCP server will reply with either a DHCPLEASEACTIVE, DHCPLEASEUNASSIGNED, or DHCPLEASEUNKNOWN message.

When the DHCP server replies to a DHCPLEASEQUERY message:

- a DHCPLEASEACTIVE message allows the DHCP relay agent to determine the IP endpoint location and the remaining duration of the IP address lease
- a DHCPLEASEUNASSIGNED message indicates that there is no current active lease for the IP address, but the DHCP server does manage that IP address
- a DHCPLEASEUNKNOWN message indicates that the DHCP server supports DHCP Leasequery but has no knowledge of the query information specified in the DHCPLEASEQUERY message (e.g. IP address, MAC address, or Client identifier option)

To enable the DHCP Leasequery feature, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp leasequery enable
```

To disable the DHCP Leasequery feature, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp leasequery enable
```

To display information about DHCP Leasequery messages, use either of the commands:

```
awplus# show counter dhcp-server
awplus# show ip dhcp server statistics
```

To display information about the current configuration of the DHCP server, including whether the DHCP server is configured to support DHCP Leasequery, use the command:

```
awplus# show ip dhcp server summary
```

Set the Options

DHCP allows clients to receive options from the DHCP server. Options describe the network configuration, and various services that are available on the network. Options are configured separately on each DHCP pool. You can configure both standard predefined options and user-defined options for a DHCP pool.

To create a user-defined option, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp option <1-254> [name <option-name>] [<option-type>]
```

To add a user-defined option to a DHCP address pool, use the command sequence:

```
awplus(config)# ip dhcp pool <pool-name>
awplus(dhcp-config)# option [<1-254>|<option-name>] <option-value>
```

It is possible to add a user-defined option with the same number as an existing pre-defined option. If this situation occurs, the user-defined option takes precedence—that is, it overrides but does not eliminate the standard option.

You can set some pre-defined options using the following commands:

To set a subnet mask (option 1) for the address pool, use the command:

```
awplus(dhcp-config)# subnet-mask <mask>
```

To add a domain name (option 15) for the address pool, use the command:

```
awplus(dhcp-config)# domain-name <domain-name>
```

To add a default router (option 3) for the address pool, use the command:

```
awplus(dhcp-config)# default-router <ip-address>
```

To add a DNS server (option 6) for the address pool, use the command:

```
awplus(dhcp-config)# dns-server <ip-address>
```

DHCP Lease Probing

Probing is used by the DHCP server to check whether an IP address it wants to lease to a client is already being used by another host. Probing is configured on a per-DHCP pool basis. You can specify probing either by ICMP Echo Request (ping) or by ARPing. ARP probing is useful in networks where ICMP may be blocked on some devices, whereas ARP is always supported. ARP and ping probing are mutually exclusive and cannot operate concurrently within a DHCP pool.

Probing is enabled by default when a DHCP pool is created.

To enable probing if probing has previously been disabled for a DHCP pool, enter the configuration mode for the pool with the **ip dhcp pool** command and then use the command:

```
awplus(dhcp-config)# probe enable
```

The default probe type is ping. To specify the probe type as ARP, enter the configuration mode for the pool and then use the command:

```
awplus(dhcp-config)# probe type arp
```

To set the timeout value in milliseconds to wait for a response after each probe packet is sent, use the command:

```
awplus(dhcp-config)# probe timeout <50-5000>
```

To specify the number of packets sent for each lease probe, use the command:

```
awplus(dhcp-config)# probe packets <0-10>
```

To disable probing for a DHCP pool, enter the configuration mode for the pool and then use the command:

```
awplus(dhcp-config)# no probe enable
```

To display the lease probe configuration settings for a specific DHCP pool or for all DHCP pools configured on the device, use the command:

```
awplus# show ip dhcp pool [<address-pool>]
```

DHCP Relay Agent Introduction

DHCP relay agents pass BOOTP messages between servers and clients. Networks where the DHCP or BOOTP server does not reside on the same IP subnet as its clients need the routers attached to the subnet to act as DHCP relay agents.

Note that both BOOTP and DHCP use BOOTP messages, allowing DHCP relay agents to relay all their packets.

Your device's DHCP Relay Agent relays these message types:

- BOOTREQUEST messages originating from any of the device's interfaces to a user-defined destination
- BOOTREPLY messages addressed to BOOTP clients on networks directly connected to the device

The DHCP relay agent ignores BOOTREPLY messages addressed to clients on networks not directly connected to the device. The device treats these as ordinary IP packets for forwarding.

A BOOTREQUEST message is relayed via unicast.

The hops field in a BOOTP message records the number of DHCP relay agents the message has been through. If the value of the hops field exceeds a predefined threshold, the DHCP relay agent discards the message.

Configuring the DHCP Relay Agent

To enable the DHCP relay agent on your device, use the commands:

```
awplus# configure terminal
awplus(config)# service dhcp-relay
```

Note DHCP relay agent is enabled by default on your switch. You only need to enter a `service dhcp-relay` command if DHCP relay agent is disabled on your switch.

You must define a relay destination on one of the device's interfaces before the relay agent can relay packets. This is the path to the DHCP server. To define a relay destination on the currently specified interface, on which there are DHCP clients, use the commands:

```
awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay server-address {<ipv4-
address>|
<ipv6-address> <server-interface>}
```

You can define more than one relay destination on your device. The following table describes how the DHCP relay agent forwards the packets.

If an interface has...	Then the relay agent relays BOOTP packets it receives on that interface to...
one relay destination defined	the relay destination.
multiple relay destinations defined	each defined relay destination.

To delete a DHCP relay destination, use the command:

```
awplus(config-if)# no ip dhcp-relay server-address {<ipv4-
                   address>|<ipv6-address> <server-interface>}
```

See the **ip dhcp-relay server-address** command on page 90.22 and the **service dhcp-relay** command on page 90.36 for command description and command examples. DHCP servers with IPv4 and IPv6 addresses are configured with **ip dhcp-relay server-address**.

When the 'hops' field in a BOOTP message exceeds a predefined threshold the BOOTP message is discarded. The default of the threshold is 10. To set the threshold, use the command:

```
awplus(config-if)# ip dhcp-relay maxhops <1-255>
```

To display the current configuration of the DHCP relay agent, use the command:

```
awplus# show ip dhcp-relay [interface <interface-name>]
```

DHCP Relay Agent Information Option (Option 82)

You can use DHCP Relay Agent Information Option (Option 82) to protect your switch from spoofing attacks, where untrusted hosts can send requests for IP addresses to access the network. The switch relays these requests to DHCP servers and the servers send IP address leases in response. Untrusted hosts could then use these IP addresses for spoofing attacks. Option 82 contains information about the location of a DHCP client for the DHCP server.

Enabling the DCHP Relay Agent Information Option feature on the switch allows the switch to insert extra information into the DHCP packets that it is relaying. This information enables accurate identification of a subscriber, as it states which interface on which relay switch the subscriber is connected to. The information is stored in an optional field in the DHCP packet header, the relay agent-information option field, with the option ID 82.

The DHCP relay agent inserts the Option 82 information into the DHCP packets that it is relaying to a DHCP server. DHCP servers that are configured to recognize Option 82 may use the information to implement IP addresses, or other parameter assignment policies, based on the network location of the client device. Alternatively, the server can simply log this information to create a detailed audit trail of the locations of the clients to which given addresses were allocated at given times.

If Option 82 insertion is enabled, then the DHCP packet flow is as follows:

- The DHCP client generates a DHCP request and broadcasts it on the network.
- The DHCP relay agent intercepts the broadcast DHCP request packet and inserts the Option 82 field in the packet.
- The DHCP relay agent unicasts the DHCP request that includes the Option 82 field to the DHCP server.
- The DHCP server receives the packet.
- If the DHCP server supports Option 82, then it echoes the Option 82 field in the DHCP reply. If the server does not support Option 82, it ignores the option and does not echo it in the reply.
- The DHCP server unicasts the reply to the relay agent.
- The relay agent removes the Option 82 field and forwards the packet to the switch port connected to the DHCP client that sent the DHCP request.

For information about DHCP Relay Agent Information Option (Option 82), see RFC 3046.

To enable the relay agent to insert its details into the Option 82 field in requests received from clients on a particular interface, use the command:

```
awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay agent-option
```

The Option 82 field contains sub-options. You can specify a value for the Remote ID sub-option, which contains information that identifies the host. To specify a value for the Remote ID, use the command:

```
awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay agent-option remote-id
<remote-id>
```

If a Remote ID value is not specified, the Remote ID sub-option is set to the switch's MAC address. You can also configure the Remote ID value as an alphanumeric string.

Note that the Option 82 agent information added by DHCP Relay differs from the information inserted by DHCP snooping (see **"DHCP Relay Agent Option 82" on page 79.4**).

Dealing with client-originated packets that already contain Option 82 information

It is possible that the requests arriving from the clients to the relay agent could already contain Option 82 data. There are two main circumstances in which this can occur:

1. A client is maliciously inserting bogus information into the packet in an attempt to subvert the process of identifying the client's location. In this case, you would want to drop the packets that contain the bogus information (or remove bogus information).

2. A Layer 2 DHCP snooping switch, that sits between the clients and the DCHP relay, is validly inserting the Option 82 information into the packets. The DHCP snooping switch is not acting as a relay agent, but is inserting the Option 82 information. In this case, you would want to forward the valid information to the DHCP server.

The action taken on packets with an Option 82 field is configurable. The command to configure this action is shown below:

```
awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay information policy
                   {append|drop|keep|replace}
```

This command sets the action that the DHCP relay takes when a received DHCP client request contains Option 82 information. Required parameters configure the switch to:

- Leave the existing Option 82 field untouched (`keep` parameter)
- Append its own Option 82 field after the existing field (`append` parameter - use this when there is a trusted DHCP Snooping switch or another relay device between the clients and the DHCP Relay)
- Drop the packet (`drop` parameter)
- Replace the existing Option 82 information with its own (the default - `replace` parameter).

See the **ip dhcp-relay information policy** command on page 90.17 for a command description and command examples.

Checking Option 82 information in DHCP server responses

To configure the switch to check for the presence of Option 82 information in response packets returned from DHCP servers, configure DHCP-relay agent-option checking with the Interface Configuration command:

```
awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay agent-option checking
```

This command enables the DHCP Relay Agent to check Option 82 information in response packets returned from DHCP servers. If the information does not match the information it has configured for its own client (downstream) interface then the DHCP relay agent drops the packet.

See the **ip dhcp-relay agent-option checking** command on page 90.15 for a command description and command examples.

Option 82 maximum message length

When a DHCP Relay (that has Option 82 insertion enabled) receives a request packet from a DHCP client, it will append the Option 82 component data, and forward the packet to the DHCP server. The DHCP client will sometimes issue packets containing pad option fields that can be overwritten with Option 82 data.

Where there are insufficient pad option fields to contain all the Option 82 data, the DHCP relay will increase the packet size to accommodate the Option 82 data. If the new (increased) packet size exceeds that defined by the **maximum-message-length** parameter, of the **ip dhcp-relay max-message-length** command then the DHCP relay will drop the packet.

```
awplus(config)# interface <interface-name>
awplus(config-if)# ip dhcp-relay max-message-length <548-1472>
```

DHCPv6 Relay Agent Notification for DHCPv6 PD

DHCPv6 relay agent notification for DHCPv6 PD (prefix delegation) allows the switch configured as a DHCPv6 relay agent to find prefix delegation options by reviewing DHCPv6 RELAY-REPLY packets that are relayed by the DHCPv6 relay agent to the DHCPv6 client.

When a DHCPv6 prefix delegation is option is found by the DHCPv6 relay agent, the DHCPv6 relay agent extracts information about the prefix being delegated to and inserts an IPv6 route matching the DHCPv6 prefix delegation information.

Future packets destined for that prefix sent through the DHCPv6 relay agent are forwarded based on information contained in the DHCPv6 prefix delegation information.

The IPv6 route is left in the routing table until the DHCPv6 prefix delegation lease time expires, or the DHCPv6 relay agent receives a release packet from the DHCPv6 client to release the DHCPv6 prefix delegation. No user configuration is required for this functionality. The DHCPv6 relay agent automatically manages IPv6 routes.

The IPv6 routes are added when the DHCPv6 relay agent relays a RELAY-REPLY packet. The DHCPv6 relay agent deletes the IPv6 routes when the DHCPv6 prefix delegation lease time expires, or the DHCPv6 relay agent receives a release packet from the DHCPv6 client. An IPv6 route in the routing table of the DHCPv6 relay agent is updated when the DHCPv6 prefix delegation lease time is extended.

This functionality leaves an IPv6 route on the routing table of the DHCPv6 relay agent. This IPv6 address allows unicast reverse packet forwarding (RPF) to work by allowing the switch to confirm that the IPv6 address on the DHCPv6 relay agent is not spoofed.

IPv6 routes are removed when a DHCP-DECLINE message is sent by the DHCPv6 client to the DHCPv6 relay agent.

Configuring the DHCP Client

You can configure an interface on your device with a static IP address, or with a dynamic IP address assigned using your device's DHCP client. When you use the DHCP client, it obtains the IP address for the interface, and other IP configuration parameters, from a DHCP server. To configure an interface and gain its IP configuration using the DHCP client, use the command:

```
awplus(config)# interface <ifname>
awplus(config-if)# ip address dhcp [client-id <interface>]
[hostname <hostname>]
```

The DHCP client supports the following IP configuration options:

- Option 1—the subnet mask for your device.
- Option 3—a list of default routers.
- Option 6—a list of DNS servers. This list appends the DNS servers set on your device with the **ip name-server** command.
- Option 15—a domain name used to resolve host names. This option replaces the domain name set with the **ip domain-name** command. Your device ignores this domain name if it has a domain list set using the **ip domain-list** command.
- Option 51—lease expiration time.

If an IP interface is configured to get its IP address and subnet mask from DHCP, the interface does not take part in IP routing until the IP address and subnet mask have been set by DHCP.

For information on configuring a static IP address on an interface, see the **ip address** command on page 29.13.

Clearing Dynamically Allocated Lease Bindings

A lease binding is the mapping of an IP address to a physical address. To clear dynamically allocated lease bindings on a DHCP server, use the command:

```
awplus# clear ip dhcp binding {ip <ip-address>|mac <mac-
address>|all|pool <pool-name>|range <low-ip-address>
<high-ip-address>}
```

You have the option to clear either a specific lease binding, specified by IP or MAC address, or to clear several lease bindings at once. The options for clearing multiple lease bindings are:

- **all**, to clear all DHCP bindings
- **pool**, to clear a specific DHCP server address pool
- **range**, to clear a range of DHCP clients

Chapter 90: Dynamic Host Configuration Protocol (DHCP) Commands

Command List 90.2
bootfile 90.2
clear ip dhcp binding 90.3
default-router 90.4
dns-server 90.5
domain-name 90.6
host 90.7
ip address dhcp 90.8
ip dhcp bootp ignore 90.9
ip dhcp leasequery enable 90.10
ip dhcp option 90.11
ip dhcp pool 90.13
ip dhcp-relay agent-option 90.14
ip dhcp-relay agent-option checking 90.15
ip dhcp-relay agent-option remote-id 90.16
ip dhcp-relay information policy 90.17
ip dhcp-relay maxhops 90.19
ip dhcp-relay max-message-length 90.20
ip dhcp-relay server-address 90.22
lease 90.24
network (DHCP) 90.26
next-server 90.27
option 90.28
probe enable 90.30
probe packets 90.31
probe timeout 90.32
probe type 90.33
range 90.34
route 90.35
service dhcp-relay 90.36
service dhcp-server 90.37
show counter dhcp-client 90.38
show counter dhcp-relay 90.39
show counter dhcp-server 90.43
show dhcp lease 90.45
show ip dhcp binding 90.46
show ip dhcp pool 90.47
show ip dhcp-relay 90.50
show ip dhcp server statistics 90.51
show ip dhcp server summary 90.53
subnet-mask 90.54

Command List

This chapter provides an alphabetical reference for commands used to configure DHCP. For more information, see **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

bootfile

This command sets the boot filename for a DHCP server pool. This is the name of the boot file that the client should use in its bootstrap process. It may need to include a path.

The **no** variant of this command removes the boot filename from a DHCP server pool.

Syntax

```
bootfile <filename>
no bootfile
```

Parameter	Description
<filename>	The boot file name.

Mode DHCP Configuration

Example To configure the boot filename for a pool P2, use the command:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# bootfile boot/main_boot.bt
```

clear ip dhcp binding

This command clears either a specific lease binding or the lease bindings specified by the command or DHCP server. The command will only take effect on dynamically allocated bindings, not statically configured bindings.

Syntax

```
clear ip dhcp binding {ip <ip-address>|mac <mac-address>|all|pool
    <pool-name>|range <low-ip-address> <high-ip-address>}
```

Parameter	Description
`ip <ip-address>`	IPv4 address of the DHCP client, in dotted decimal notation in the format A.B.C.D.
`mac <mac-address>`	MAC address of the DHCP client, in hexadecimal notation in the format HHHH.HHHH.HHHH.
`all`	All DHCP bindings.
`pool <pool-name>`	Description used to identify DHCP server address pool. Valid characters are any printable character. If the name contains spaces then you must enclose these in "quotation marks".

`range`
`<low-ip-address>`
`<high-ip-address>`
	IPv4 address range for DHCP clients, in dotted decimal notation. The first IP address is the low end of the range, the second IP address is the high end of the range.

Mode User Exec and Privileged Exec

Usage A specific binding may be deleted by **ip** address or **mac** address, or several bindings may be deleted at once using **all**, **pool** or **range**.

Note that if you specify to clear the **ip** or **mac** address of what is actually a static DHCP binding, an error message is displayed. If **all**, **pool** or **range** are specified and one or more static DHCP bindings exist within those addresses, any dynamic entries within those addresses are cleared but any static entries are not cleared.

Examples To clear the specific IP address binding `192.168.1.1`, use the command:

```
awplus# clear ip dhcp binding ip 192.168.1.1
```

To clear all dynamic DHCP entries, use the command:

```
awplus# clear ip dhcp binding all
```

Related Commands **show ip dhcp binding**

default-router

This command adds a default router to the DHCP address pool you are configuring. You can use this command multiple times to create a list of default routers on the client's subnet. This sets the router details using the pre-defined option 3. Note that if you add a user-defined option 3 using the **option** command, then you will override any settings created with this command.

The **no** variant of this command removes either the specified default router, or all default routers from the DHCP pool.

Syntax

```
default-router <ip-address>
no default-router [<ip-address>]
```

Parameter	Description
<ip-address>	IPv4 address of the default router, in dotted decimal notation.

Mode DHCP Configuration

Examples To add a router with an IP address 192.168.1.2 to the DHCP pool named P2, use the following commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# default-router 192.168.1.2
```

To remove a router with an IP address 192.168.1.2 to the DHCP pool named P2, use the following commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no default-router 192.168.1.2
```

To remove all routers from the DHCP pool named P2, use the following commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no default-router
```

dns-server

This command adds a Domain Name System (DNS) server to the DHCP address pool you are configuring. You can use this command multiple times to create a list of DNS name servers available to the client. This sets the DNS server details using the pre-defined option 6. Note that if you add a user-defined option 6 using the **option** command on page 90.28, then you will override any settings created with this command.

The **no** variant of this command removes either the specified DNS server, or all DNS servers from the DHCP pool.

Syntax

```
dns-server <ip-address>
no dns-server [<ip-address>]
```

Parameter	Description
<ip-address>	IPv4 address of the DNS server, in dotted decimal notation.

Mode DHCP Configuration

Examples To add the DNS server with the assigned IP address 192.168.1.1 to the DHCP pool named P1, use the following commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# dns-server 192.168.1.1
```

To remove the DNS server with the assigned IP address 192.168.1.1 from the DHCP pool named P1, use the following commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no dns-server 192.168.1.1
```

To remove all DNS servers from the DHCP pool named P1, use the following commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no dns-server
```

Related Commands

default-router
option
service dhcp-server
show ip dhcp pool
subnet-mask

domain-name

This command adds a domain name to the DHCP address pool you are configuring. Use this command to specify the domain name that a client should use when resolving host names using the Domain Name System. This sets the domain name details using the pre-defined option 15. Note that if you add a user-defined option 15 using the **option** command on page 90.28, then you will override any settings created with this command.

The **no** variant of this command removes the domain name from the address pool.

Syntax

```
domain-name <domain-name>
no domain-name
```

Parameter	Description
<domain-name>	The domain name you wish to assign the DHCP pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”.

Mode DHCP Configuration

Examples To add the domain name `Nerv_Office` to DHCP pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# domain-name Nerv_Office
```

To remove the domain name `Nerv_Office` from DHCP pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no domain-name Nerv_Office
```

Related Commands

default-router
dns-server
option
service dhcp-server
show ip dhcp pool
subnet-mask

host

This command adds a static host address to the DHCP address pool you are configuring. The client with the matching MAC address is permanently assigned this IP address. No other clients can request it.

The **no** variant of this command removes the specified host address from the DHCP pool. Use the **no host all** command to remove all static host addresses from the DHCP pool.

Syntax

```
host <ip-address> <mac-address>
no host <ip-address>
no host all
```

Parameter	Description
<ip-address>	IPv4 address of the DHCP client, in dotted decimal notation in the format A.B.C.D
<mac-address>	MAC address of the DHCP client, in hexadecimal notation in the format HHHH.HHHH.HHHH

Mode DHCP Configuration

Usage Note that a network/mask must be configured using a **network** command before issuing a **host** command. Also note that a host address must match a network to add a static host address.

Examples To add the host at `192.168.1.5` with the MAC address `000a.451d.6e34` to DHCP pool `1`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool 1
awplus(dhcp-config)# network 192.168.1.0/24
awplus(dhcp-config)# host 192.168.1.5 000a.451d.6e34
```

To remove the host at `192.168.1.5` with the MAC address `000a.451d.6e34` from DHCP pool `1`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool 1
awplus(dhcp-config)# no host 192.168.1.5 000a.451d.6e34
```

Related Commands
lease
range
show ip dhcp pool

ip address dhcp

This command activates the DHCP client on the interface you are configuring. This allows the interface to use the DHCP client to obtain its IP configuration details from a DHCP server on its connected network.

The **client-id** and **hostname** parameters are identifiers that you may want to set in order to interoperate with your existing DHCP infrastructure. If neither option is needed, then the DHCP server uses the MAC address field of the request to identify the host.

The DHCP client supports the following IP configuration options:

- Option 1 - the subnet mask for your device.
- Option 3 - a list of default routers.
- Option 6 - a list of DNS servers. This list appends the DNS servers set on your device with the ip name-server command.
- Option 15 - a domain name used to resolve host names. This option replaces the domain name set with the ip domain-name command. Your device ignores this domain name if it has a domain list set using the ip domain-list command.
- Option 51 - lease expiration time.

The **no** variant of this command stops the interface from obtaining IP configuration details from a DHCP server.

Syntax

```
ip address dhcp [client-id <interface>] [hostname <hostname>]
no ip address dhcp
```

Parameter	Description
<interface>	The name of the interface you are activating the DHCP client on. If you specify this, then the MAC address associated with the specified interface is sent to the DHCP server in the optional identifier field. Default: no default
<hostname>	The hostname for the DHCP client on this interface. Typically this name is provided by the ISP. Default: no default

Mode Interface Configuration for a VLAN interface.

Examples To set the interface `vlan10` to use DHCP to obtain an IP address, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# ip address dhcp
```

To stop the interface `vlan10` from using DHCP to obtain its IP address, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# no ip address dhcp
```

Related Commands ip address

Validation Commands show running-config
show ip interface

ip dhcp bootp ignore

This command configures the DHCP server to ignore any BOOTP requests it receives. The DHCP server accepts BOOTP requests by default.

The **no** variant of this command configures the DHCP server to accept BOOTP requests. This is the default setting.

Syntax

```
ip dhcp bootp ignore
no ip dhcp bootp ignore
```

Mode Global Configuration

Examples To configure the DHCP server to ignore BOOTP requests, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp bootp ignore
```

To configure the DHCP server to respond to BOOTP requests, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp bootp ignore
```

Related Commands show ip dhcp server summary

ip dhcp leasequery enable

Use this command to enable the DHCP server to respond to DHCPLEASEQUERY packets. Enabling the DHCP leasequery feature allows a DHCP Relay Agent to obtain IP address information directly from the DHCP server using DHCPLEASEQUERY messages.

Use the **no** variant of this command to disable the support of DHCPLEASEQUERY packets.

For more information, see **"Enable DHCP Leasequery" on page 89.5**.

Syntax

```
ip dhcp leasequery enable
no ip dhcp leasequery enable
```

Default DHCP leasequery support is disabled by default.

Mode Global Configuration

Examples To enable DHCP leasequery support, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp leasequery enable
```

To disable DHCP leasequery support, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp leasequery enable
```

Related Commands

show counter dhcp-server
show ip dhcp server statistics
show ip dhcp server summary

ip dhcp option

This command creates a user-defined DHCP option. You can then use this option when configuring a DHCP pool, by using the **option** command. Options with the same number as one of the pre-defined options override the standard option definition. The pre-defined options use the option numbers 1, 3, 6, 15, and 51.

The **no** variant of this command removes either the specified user-defined option, or removes all user-defined options. This also automatically removes the user-defined options from the associated DHCP address pools.

Syntax

```
ip dhcp option <1-254> [name <option-name>] [<option-type>]
no ip dhcp option [<1-254>|<option-name>]
```

Parameter	Description	
<1-254>	The option number of the option. Options with the same number as one of the standard options overrides the standard option definition.	
<option-name>	Option name used to identify the option. You cannot use a number as the option name. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”. Default: no default	
<option-type>	The option value. You must specify a value that is appropriate to the option type:	
	`asci`	An ASCII text string
	`hex`	A hexadecimal string. Valid characters are the numbers 0–9 and letters a–f. Embedded spaces are not valid. The string must be an even number of characters, from 2 and 256 characters long.
	`ip`	An IPv4 address or mask that has the dotted decimal A.B.C.D notation. To create a list of IP addresses, you must add each IP address individually by using the option command multiple times.
	`integer`	A number from 0 to 4294967295.
	`flag`	A value that either sets (to 1) or unsets (to 0) a flag: **true**, **on**, or **enabled** will set the flag. **false**, **off** or **disabled** will unset the flag.

Mode Global Configuration

Examples To define a user-defined ASCII string option as option 66, without a name, use the command:

```
awplus# configure terminal
awplus(config)# ip dhcp option 66 ascii
```

To define a user-defined hexadecimal string option as option 46, with the name "tcpip-node-type", use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp option 46 name tcpip-node-type hex
```

To define a user-defined IP address option as option 175, with the name `special-address`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp option 175 name special-address ip
```

To remove the specific user-defined option with the option number `12`, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp option 12
```

To remove the specific user-defined option with the option name `perform-router-discovery`, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp option perform-router-discovery
```

To remove all user-defined option definitions, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp option
```

Related Commands

default-router
dns-server
domain-name
option
service dhcp-server
show ip dhcp server summary
subnet-mask

ip dhcp pool

This command will enter the configuration mode for the pool name specified. If the name specified is not associated with an existing pool, the switch will create a new pool with this name, then enter the configuration mode for the new pool.

Once you have entered the DHCP configuration mode, all commands executed before the next **exit** command will apply to this pool.

You can create multiple DHCP pools on devices with multiple interfaces. This allows the device to act as a DHCP server on multiple interfaces to distribute different information to clients on the different networks.

The **no** variant of this command deletes the specific DHCP pool.

Syntax

```
ip dhcp pool <pool-name>
no ip dhcp pool <pool-name>
```

Parameter	Description
<pool-name>	Description used to identify this DHCP pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”.

Mode Global Configuration

Example To create the DHCP pool named `P2` and enter DHCP Configuration mode, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)#
```

To delete the DHCP pool named `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# no ip dhcp pool P2
```

Related Commands **service dhcp-server**

ip dhcp-relay agent-option

This command enables the DHCP Relay Agent to insert the DHCP Relay Agent Information Option (Option 82) into the client-request packets that it relays to its DHCP server. This allows the DHCP Relay Agent to pass on information to the server about the network location of the client device. The DHCP Relay Agent strips the DHCP Relay Agent Option 82 field out of the DHCP server's response, so that the DHCP client never sees this field.

When the DHCP Relay Agent appends its DHCP Relay Agent Option 82 data into the packet, it first overwrites any pad options present; then if necessary, it increases the packet length to accommodate the DHCP Relay Agent Option 82 data.

The **no** variant of this command stops the DHCP Relay Agent from appending the Option 82 field onto DHCP requests before forwarding it to the server.

For DHCP Relay Agent and DHCP Relay Agent Option 82 introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. and **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**

Note The DHCP-relay service might alter the content of the DHCP Relay Agent Option 82 field, if the commands **ip dhcp-relay agent-option** and **ip dhcp-relay information policy** have been configured.

Syntax

```
ip dhcp-relay agent-option
no ip dhcp-relay agent-option
```

Default DHCP Relay Agent Information Option (Option 82) insertion is disabled by default.

Mode Interface Configuration for a VLAN interface.

Usage Use this command to alter the DHCP Relay Agent Option 82 setting when your switch is the first hop for the DHCP client. To limit the maximum length of the packet, use the **ip dhcp-relay max-message-length** command.

This command cannot be enabled if DHCP snooping is enabled on your switch (**service dhcp-snooping** command on page 80.23), and vice versa.

Examples To make the DHCP Relay Agent listening on `vlan15` append the DHCP Relay Agent Option 82 field, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan15
awplus(config-if)# ip dhcp-relay agent-option
```

To stop the DHCP Relay Agent from appending the DHCP Relay Agent Option 82 field on `vlan15`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan15
awplus(config-if)# no ip dhcp-relay agent-option
```

Related Commands
ip dhcp-relay agent-option remote-id
ip dhcp-relay information policy
ip dhcp-relay max-message-length
service dhcp-relay

ip dhcp-relay agent-option checking

This command enables the DHCP Relay Agent to check DHCP Relay Agent Information Option (Option 82) information in response packets returned from DHCP servers. If the information does not match the information it has for its own client (downstream) interface then the DHCP Relay Agent drops the packet. Note that **ip dhcp-relay agent-option** must be configured.

The DHCP Relay Agent Option 82 field is included in relayed client DHCP packets if:

- DHCP Relay Agent Option 82 is enabled (**ip dhcp-relay agent-option**), and
- DHCP Relay Agent is enabled on the switch (**service dhcp-relay**)

For DHCP Relay Agent and DHCP Relay Agent Option 82 introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. and **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**

Syntax
```
ip dhcp-relay agent-option checking
no ip dhcp-relay agent-option checking
```

Mode Interface Configuration for a VLAN interface.

Examples To make the DHCP Relay Agent listening on `vlan10` check the DHCP Relay Agent Information Option (Option 82) field, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# ip dhcp-relay agent-option
awplus(config-if)# ip dhcp-relay agent-option checking
```

To stop the DHCP Relay Agent on `vlan10` from checking the DHCP Relay Agent Information Option (Option 82) field, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# no ip dhcp-relay agent-option checking
```

Related Commands
ip dhcp-relay agent-option
ip dhcp-relay agent-option remote-id
ip dhcp-relay information policy
service dhcp-relay

ip dhcp-relay agent-option remote-id

Use this command to specify the Remote ID sub-option of the DHCP Relay Agent Option 82 field the DHCP Relay Agent inserts into clients' request packets. The Remote ID identifies the device that is inserting the DHCP Relay Agent Option 82 information. If a Remote ID is not specified, the Remote ID sub-option is set to the switch's MAC address.

Use the **no** variant of this command to return the Remote ID for an interface.

For DHCP Relay Agent and DHCP Relay Agent Option 82 introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. and **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**

Syntax

```
ip dhcp-relay agent-option remote-id <remote-id>
no ip dhcp-relay agent-option remote-id
```

Parameter	Description
<remote-id>	An alphanumeric (ASCII) string, 1 to 63 characters in length. Additional characters allowed are hyphen (-), underscore (_) and hash (#). Spaces are not allowed.

Default The Remote ID is set to the switch's MAC address by default.

Mode Interface Configuration for a VLAN interface.

Usage The Remote ID sub-option is included in the DHCP Relay Agent Option 82 field of relayed client DHCP packets if:

- DHCP Relay Agent Option 82 is enabled (**ip dhcp-relay agent-option**), and
- DHCP Relay Agent is enabled on the switch (**service dhcp-relay**)

Examples To set the Remote ID to `myid` for client DHCP packets received on `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ip dhcp-relay agent-option remote-id myid
```

To remove the Remote ID specified for `vlan1`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# no ip dhcp-relay agent-option remote-id
```

Related Commands

ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
show ip dhcp-relay

ip dhcp-relay information policy

This command sets the policy for how the DHCP relay deals with packets arriving from the client that contain DHCP Relay Agent Option 82 information.

If the command **ip dhcp-relay agent-option** has not been configured, then this command has no effect at all - no alteration is made to Option 82 information in packets arriving from the client side.

However, if the command **ip dhcp-relay agent-option** has been configured, this command modifies how the DHCP relay service deals with cases where the packet arriving from the client side already contains DHCP Relay Agent Option 82 information.

This command sets the action that the DHCP relay should take when a received DHCP client request contains DHCP Relay Agent Option 82 information.

By default, the DHCP Relay Agent replaces any existing DHCP Relay Agent Option 82 field with its own DHCP Relay Agent field. This is equivalent to the functionality of the **replace** parameter.

The **no** variant of this command returns the policy to the default behavior - i.e. replacing the existing DHCP Relay Agent Option 82 field.

For DHCP Relay Agent and DHCP Relay Agent Option 82 introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. and **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**

Note

The DHCP-relay service might alter the content of the DHCP Relay Agent Option 82 field, if the commands **ip dhcp-relay agent-option** and **ip dhcp-relay information policy** have been configured.

Syntax

```
ip dhcp-relay information policy {append|drop|keep|replace}

no ip dhcp-relay information policy
```

Parameter	Description
`append`	The DHCP Relay Agent appends the DHCP Relay Agent Option 82 field of the packet with its own DHCP Relay Agent Option 82 details.
`drop`	The DHCP Relay Agent discards the packet.
`keep`	The DHCP Relay Agent forwards the packet without altering the DHCP Relay Agent Option 82 field.
`replace`	The DHCP Relay Agent replaces the existing DHCP Relay Agent details in the DHCP Relay Agent Option 82 field with its own details before forwarding the packet.

Mode Interface Configuration for a VLAN interface.

Examples To make the DHCP Relay Agent listening on `vlan15` drop any client requests that already contain DHCP Relay Agent Option 82 information, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan15
awplus(config-if)# ip dhcp-relay information policy drop
```

To reset the DHCP relay information policy to the default policy for interface `vlan15`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan15
awplus(config-if)# no ip dhcp-relay information policy
```

Related Commands

ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
service dhcp-server

ip dhcp-relay maxhops

This command sets the hop count threshold for discarding BOOTP messages. When the hops field in a BOOTP message exceeds the threshold, the DHCP Relay Agent discards the BOOTP message. The hop count threshold is set to 10 hops by default.

Use the **no** variant of this command to reset the hop count to the default.

For DHCP Relay Agent introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. For Option 82 introductory information, see **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**.

Syntax

```
ip dhcp-relay maxhops <1-255>
no ip dhcp-relay maxhops
```

Parameter	Description
<1-255>	The maximum hop count value.

Default The default hop count threshold is 10 hops.

Mode Interface Configuration for a VLAN interface.

Example To set the maximum number of hops to 5 for packets received on interface `vlan15`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan15
awplus(config-if)# ip dhcp-relay maxhops 5
```

Related Commands **service dhcp-relay**

ip dhcp-relay max-message-length

This command applies when the switch is acting as a DHCP Relay Agent and DHCP Relay Agent Option 82 insertion is enabled. It sets the maximum DHCP message length (in bytes) for the DHCP packet with its DHCP Relay Agent Option 82 data inserted. From this value it calculates the maximum packet size that it will accept at its input. Packets that arrive greater than this value will be dropped.

The **no** variant of this command sets the maximum message length to its default of 1400 bytes.

For DHCP Relay Agent introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. For Option 82 introductory information, see **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**.

Syntax

```
ip dhcp-relay max-message-length <548-1472>
no ip dhcp-relay max-message-length
```

Parameter	Description
<548-1472>	The maximum DHCP message length (this is the message header plus the inserted DHCP option fields in bytes).

Default The default is 1400 bytes.

Mode Interface Configuration for a VLAN interface.

Usage When a DHCP Relay Agent (that has DHCP Relay Agent Option 82 insertion enabled) receives a *request* packet from a DHCP client, it will append the DHCP Relay Agent Option 82 component data, and forward the packet to the DHCP server. The DHCP client will sometimes issue packets containing pad option fields that can be overwritten with Option 82 data.

Where there are insufficient pad option fields to contain all the DHCP Relay Agent Option 82 data, the DHCP Relay Agent will increase the packet size to accommodate the DHCP Relay Agent Option 82 data. If the new (increased) packet size exceeds that defined by the **maximum-message-length** parameter, then the DHCP Relay Agent will drop the packet.

Note Before setting this command, you must first run the **ip dhcp-relay agent-option** command on page 90.14. This will allow the DHCP Relay Agent Option 82 fields to be appended.

Example To set the maximum DHCP message length to 1200 bytes for packets arriving in interface `vlan7`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan7
awplus(config-if)# ip dhcp-relay max-message-length 1200
```

To reset the maximum DHCP message length to the default of 1400 bytes for packets arriving in interface `vlan7`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan7
awplus(config-if)# no ip dhcp-relay max-message-length
```

Related Commands **service dhcp-relay**

ip dhcp-relay server-address

This command adds a DHCP server for the DHCP Relay Agent to forward client DHCP packets to on a particular interface. You can add up to five DHCP servers on each device interface that the DHCP Relay Agent is listening on.

The **no** variant of this command deletes the specified DHCP server from the list of servers available to the DHCP relay agent.

For DHCP Relay Agent introductory information, see **"DHCP Relay Agent Introduction" on page 89.9**. For Option 82 introductory information, see **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**.

Syntax

```
ip dhcp-relay server-address {<ipv4-address>|
    <ipv6-address> <server-interface>}

no ip dhcp-relay server-address {<ipv4-address>|
    <ipv6-address> <server-interface>}
```

Parameter	Description
<ipv4-address>	Specify the IPv4 address of the DHCP server for DHCP Relay Agent to forward client DHCP packets to, in dotted decimal notation. The IPv4 address uses the format A.B.C.D.
<ipv6-address>	Specify the IPv6 address of the DHCPv6 server for DHCPv6 Relay Agent to forward client DHCP packets to, in hexadecimal notation.
<server-interface>	Specify the interface name of the DHCPv6 server. It is only required for a DHCPv6 server with an IPv6 address.

Mode Interface Configuration for a VLAN interface.

Usage For a DHCP server with an IPv6 address you must specify the interface for the DHCP server. See examples below for configuration differences between IPv4 and IPv6 DHCP relay servers.

See also the **service dhcp-relay** command to enable the DHCP Relay Agent on your device. The **ip dhcp-relay server-address** command defines a relay destination on an interface on the device, needed by the DHCP Relay Agent to relay DHCP client packets to a DHCP server.

Examples To enable the DHCP Relay Agent to relay DHCP packets on interface `vlan2` to the DHCP server with the IPv4 address `192.0.2.200`, use the commands:

```
awplus# configure terminal
awplus(config)# service dhcp-relay
awplus(config)# interface vlan2
awplus(config-if)# ip dhcp-relay server-address 192.0.2.200
```

To remove the DHCP server with the IPv4 address `192.0.2.200` from the list of servers

available to the DHCP Relay Agent on interface `vlan2`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ip dhcp-relay server-address 192.0.2.200
```

To enable the DHCP Relay Agent on your device to relay DHCP packets on interface `vlan10` to the DHCP server with the IPv6 address `2001:0db8:010d::1` on interface `vlan20`, use the commands:

```
awplus# configure terminal
awplus(config)# service dhcp-relay
awplus(config)# interface vlan10
awplus(config-if)# ip dhcp-relay server-address
2001:0db8:010d::1 vlan20
```

To remove the DHCP server with the IPv6 address `2001:0db8:010d::1` on interface `vlan20` from the list of servers available to the DHCP Relay Agent on interface `vlan10`, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# no ip dhcp-relay server-address
2001:0db8:010d::1 vlan20
```

lease

This command sets the expiration time for a leased address for the DHCP address pool you are configuring. The time set by the days, hours, minutes and seconds is cumulative. The minimum total lease time that can be configured is 20 seconds. The maximum total lease time that can be configured is 120 days.

Note that if you add a user-defined option 51 using the **option** command, then you will override any settings created with this command. Option 51 specifies a lease time of 1 day.

Use the **infinite** parameter to set the lease expiry time to infinite (leases never expire).

Use the **no** variant of this command to return the lease expiration time back to the default of one day.

Syntax

```
lease <days> <hours> <minutes> [<seconds>]
lease infinite
no lease
```

Parameter	Description
<days>	The number of days, from 0 to 120, that the lease expiry time is configured for. Default: 1
<hours>	The number of hours, from 0 to 24, that the lease expiry time is configured for. Default: 0
<minutes>	The number of minutes, from 0 to 60, the lease expiry time is configured for. Default: 0
<seconds>	The number of seconds, from 0 to 60, the lease expiry time is configured for.
`infinite`	The lease never expires.

Default The default lease time is 1 day.

Mode DHCP Configuration

Examples To set the lease expiration time for address pool `P2` to 35 minutes, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# lease 0 0 35
```

To set the lease expiration time for the address pool `Nerv_Office` to 1 day, 5 hours, and 30 minutes, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool Nerv_Office
awplus(dhcp-config)# lease 1 5 30
```

To set the lease expiration time for the address pool `P3` to 20 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P3
awplus(dhcp-config)# lease 0 0 0 20
```

To set the lease expiration time for the pool to never expire, use the command:

```
awplus(dhcp-config)# lease infinite
```

To return the lease expiration time to the default of one day, use the command:

```
awplus(dhcp-config)# no lease
```

Related Commands

option
service dhcp-server

network (DHCP)

This command sets the network (subnet) that the DHCP address pool applies to.

The **no** variant of this command removes the network (subnet) from the DHCP address pool.

Syntax

```
network {<ip-subnet-address/prefix-length>|<ip-subnet-address/mask>}
no network
```

Parameter	Description
<ip-subnet-address/prefix-length>	The IPv4 subnet address in dotted decimal notation followed by the prefix length in slash notation.
<ip-subnet-address/mask>	The IPv4 subnet address in dotted decimal notation followed by the subnet mask in dotted decimal notation.

Mode DHCP Configuration

Usage This command will fail if it would make existing ranges invalid. For example, if they do not lie within the new network you are configuring.

The **no** variant of this command will fail if ranges still exist in the pool. You must remove all ranges in the pool before issuing a **no network** command to remove a network from the pool.

Examples To configure a network for the address pool `P2`, where the subnet is `192.0.2.5` and the mask is `255.255.255.0`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# network 192.0.2.5/24
```

or you can use dotted decimal notation instead of slash notation for the subnet-mask:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# network 192.0.2.5 255.255.255.0
```

Related Commands **service dhcp-server**
subnet-mask

next-server

This command sets the next server address for a DHCP server pool. It is the address of the next server that the client should use in its bootstrap process.

The **no** variant of this command removes the next server address from the DHCP address pool.

Syntax

```
next-server <ip-address>
no next-server
```

Parameter	Description
<ip-address>	The server IP address, entered in dotted decimal notation.

Mode DHCP Configuration

Example To set the next-server address for the address pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# next-server 192.0.2.2
```

option

This command adds a user-defined option to the DHCP address pool you are configuring. For the **hex**, **integer**, and **flag** option types, if the option already exists, the new option overwrites the existing option's value. Options with an **ip** type can hold a list of IP addresses or masks (i.e. entries that have the A.B.C.D address format), so if the option already exists in the pool, then the new IP address is added to the list of existing IP addresses.

Options with the same number as one of the pre-defined options override the standard option definition. The pre-defined options use the option numbers 1, 3, 6, 15, and 51.

The **no** variant of this command removes the specified user-defined option from the DHCP pool, or all user-defined options from the DHCP pool.

Syntax

```
option [<1-254>|<option-name>] <option-value>
no option [<1-254>|<option-value>]
```

Parameter	Description	
<1-254>	The option number of the option. Options with the same number as one of the standard options overrides the standard option definition.	
<option-name>	Option name associated with the option.	
<option-value>	The option value. You must specify a value that is appropriate to the option type:	
	`hex`	A hexadecimal string. Valid characters are the numbers 0–9 and letters a–f. Embedded spaces are not valid. The string must be an even number of characters, from 2 and 256 characters long.
	`ip`	An IPv4 address or mask that has the dotted decimal A.B.C.D notation. To create a list of IP addresses, you must add each IP address individually using the option command multiple times.
	`integer`	A number from 0 to 4294967295.
	`flag`	A value of either true, on, or enabled to set the flag, or false, off or disabled to unset the flag.

Mode DHCP Configuration

Examples To add the ASCII-type option named `tftp-server-name` to the pool `P2` and give the option the value `server1`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# option tftp-server-name server1
```

To add the hex-type option named `tcpip-node-type` to the pool `P2` and give the option the value `08af`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# option tcpip-node-type 08af
```

To add multiple IP addresses for the ip-type option `175`, use the command:

```
awplus(dhcp-config)# option 175 192.0.2.6
awplus(dhcp-config)# option 175 192.0.2.12
awplus(dhcp-config)# option 175 192.0.2.33
```

To add the option `179` to a pool, and give the option the value `123456`, use the command:

```
awplus(dhcp-config)# option 179 123456
```

To add a user-defined flag option with the name `perform-router-discovery`, use the command:

```
awplus(dhcp-config)# option perform-router-discovery yes
```

To clear all user-defined options from a DHCP address pool, use the command:

```
awplus(dhcp-config)# no option
```

To clear a user-defined option, named `tftp-server-name`, use the command:

```
awplus(dhcp-config)# no option tftp-server-name
```

Related Commands

dns-server
ip dhcp option
lease
service dhcp-server
show ip dhcp pool

probe enable

Use this command to enable lease probing for a DHCP pool. Probing is used by the DHCP server to check if an IP address it wants to lease to a client is already being used by another host.

The **no** variant of this command disables probing for a DHCP pool.

Syntax

```
probe enable
no probe enable
```

Default Probing is enabled by default.

Mode DHCP Pool Configuration

Examples To enable probing for pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# probe enable
```

To disable probing for pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe enable
```

Related Commands

ip dhcp pool
probe packets
probe timeout
probe type
show ip dhcp pool

probe packets

Use this command to specify the number of packets sent for each lease probe. Lease probing is configured on a per-DHCP pool basis. When set to 0 probing is effectively disabled.

The **no** variant of this command sets the number of probe packets sent to the default of 5.

Syntax

```
probe packets <0-10>
no probe packets
```

Parameter	Description
<0-10>	The number of probe packets sent.

Default The default is 5.

Mode DHCP Pool Configuration

Examples To set the number of probe packets to 2 for pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# probe packets 2
```

To set the number of probe packets to the default 5 for pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe packets
```

Related Commands

probe enable
probe timeout
probe type
show ip dhcp pool

probe timeout

Use this command to set the timeout value in milliseconds that the server waits for a response after each probe packet is sent. Lease probing is configured on a per-DHCP pool basis.

The **no** variant of this command sets the probe timeout value to the default setting, 200 milliseconds.

Syntax

```
probe timeout <50-5000>
no probe timeout
```

Parameter	Description
<50-5000>	Timeout interval in milliseconds.

Default The default timeout interval is 200 milliseconds.

Mode DHCP Pool Configuration

Examples To set the probe timeout value to 500 milliseconds for pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# probe timeout 500
```

To set the probe timeout value for pool `P2` to the default, 200 milliseconds, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe timeout
```

Related Commands

probe enable
probe packets
probe type
show ip dhcp pool

probe type

Use this command to set the probe type for a DHCP pool. The probe type specifies how the DHCP server checks whether an IP address is being used by other hosts, referred to as lease probing. If **arp** is specified, the server sends an ARP request to determine if an address is in use. If **ping** is specified, the server will send an ICMP Echo Request (ping).

The **no** variant of this command sets the probe type to the default setting, ping.

Syntax

```
probe type {arp|ping}
no probe type
```

Parameter	Description
`arp`	Probe using ARP.
`ping`	Probe using ping.

Default The default probe type is ping.

Mode DHCP Pool Configuration

Examples To set the probe type to `arp` for the pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# probe type arp
```

To set the probe type for the pool `P2` to the default, `ping`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no probe type
```

Related Commands

ip dhcp pool
probe enable
probe packets
probe timeout
show ip dhcp pool

range

This command adds an address range to the DHCP address pool you are configuring. The DHCP server responds to client requests received from the pool's network. It assigns an IP addresses within the specified range. The IP address range must lie within the network. You can add multiple address ranges and individual IP addresses for a DHCP pool by using this command multiple times.

The **no** variant of this command removes an address range from the DHCP pool. Use the **no range all** command to remove all address ranges from the DHCP pool.

Syntax

```
range <ip-address> [<ip-address>]
no range <ip-address> [<ip-address>]
no range all
```

Parameter	Description
<ip-address>	IPv4 address range for DHCP clients, in dotted decimal notation. The first IP address is the low end of the range, the second IP address is the high end. Specify only one IP address to add an individual IP address to the address pool.

Mode DHCP Configuration

Examples To add an address range of `192.0.2.5` to `192.0.2.16` to the pool `Nerv_Office`, use the command:

```
awplus# configure terminal
awplus(config)# ip dhcp pool Nerv_Office
awplus(dhcp-config)# range 192.0.2.5 192.0.2.16
```

To add the individual IP address `192.0.2.2` to a pool, use the command:

```
awplus(dhcp-config)# range 192.0.2.2
```

To remove all address ranges from a pool, use the command:

```
awplus(dhcp-config)# no range all
```

Related Commands

ip dhcp pool
service dhcp-server
show ip dhcp pool

route

This command allows the DHCP server to provide static routes to clients.

Syntax `route A.B.C.D/M A.B.C.D {both|opt249|rfc3442}`

Parameter	Description
`A.B.C.D/M`	Subnet for the route
`A.B.C.D`	Next hop for the route
`both`	opt249 and rft3442
`opt249`	Classless static route option for DHCP
`rfc3442`	Classless static route option for DHCP

Mode DHCP Configuration

Examples To distribute static routes for route `0.0.0.0/0` whose next hop is `192.16.1.1` to clients using both `opt249` and `rfc3442`, use the command:

```
awplus# configure terminal
awplus(config)# ip dhcp pool pubic
awplus(dhcp-config)# route 0.0.0.0/0 192.16.1.1 both
```

Related Commands **ip dhcp pool**

service dhcp-relay

This command enables the DHCP Relay Agent on the device. However, on a given IP interface, no DHCP forwarding takes place until at least one DHCP server is specified to forward/relay all clients' DHCP packets to.

The **no** variant of this command disables the DHCP Relay Agent on the device for all interfaces.

Syntax

```
service dhcp-relay
no service dhcp-relay
```

Mode Global Configuration

Usage A maximum number of 400 DHCP Relay Agents (one per interface) can be configured on the device. Once this limit has been reached, any further attempts to configure DHCP Relay Agents will not be successful.

Default The DHCP-relay service is enabled by default.

Examples To enable the DHCP relay global function, use the commands:

```
awplus# configure terminal
awplus(config)# service dhcp-relay
```

To disable the DHCP relay global function, use the commands:

```
awplus# configure terminal
awplus(config)# no service dhcp-relay
```

Related Commands

ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
ip dhcp-relay information policy
ip dhcp-relay maxhops
ip dhcp-relay server-address

service dhcp-server

This command enables the DHCP server on your device. The server then listens for DHCP requests on all IP interfaces. It will not run if there are no IP interfaces configured.

The **no** variant of this command disables the DHCP server.

Syntax

```
service dhcp-server
no service dhcp-server
```

Mode Global Configuration

Example To enable the DHCP server, use the commands:

```
awplus# configure terminal
awplus(config)# service dhcp-server
```

Related Commands

ip dhcp pool
show ip dhcp server summary
subnet-mask

show counter dhcp-client

This command shows counters for the DHCP client on your device.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show counter dhcp-client`

Mode User Exec and Privileged Exec

Example To display the message counters for the DHCP client on your device, use the command:

```
awplus# show counter dhcp-client
```

Output **Figure 90-1: Example output from the show counter dhcp-client command**

```
show counter dhcp-client

DHCPDISCOVER out      ......... 10
DHCPREQUEST out       ......... 34
DHCPDECLINE out       ......... 4
DHCPRELEASE out       ......... 0
DHCPOFFER in          ......... 22
DHCPACK in            ......... 18
DHCPNAK in            ......... 0
```

Table 90-1: Parameters in the output of the show counter dhcp-client command

Parameter	Description
DHCPDISCOVER out	The number of DHCP Discover messages sent by the client.
DHCPREQUEST out	The number of DHCP Request messages sent by the client.
DHCPDECLINE out	The number of DHCP Decline messages sent by the client.
DHCPRELEASE out	The number of DHCP Release messages sent by the client.
DHCPOFFER in	The number of DHCP Offer messages received by the client.
DHCPACK in	The number of DHCP Acknowledgement messages received by the client.
DHCPNAK in	The number of DHCP Negative Acknowledgement messages received by the client.

Related Commands ip address dhcp

show counter dhcp-relay

This command shows counters for the DHCP Relay Agent on your device.

For information on output options, see "Controlling "show" Command Output" on page 1.37.

Syntax `show counter dhcp-relay`

Mode User Exec and Privileged Exec

Examples To display counters for the DHCP Relay Agent on your device, use the following command:

```
awplus# show counter dhcp-relay
```

Output **Figure 90-2: Example output from the show counter dhcp-relay command**

```
awplus#show counter dhcp-relay

DHCP relay counters
Requests In               ......... 4
Replies In                ......... 4
Relayed To Server         ......... 4
Relayed To Client         ......... 4
Out To Server Failed      ......... 0
Out To Client Failed      ......... 0
Invalid hlen              ......... 0
Bogus giaddr              ......... 0
Corrupt Agent Option      ......... 0
Missing Agent Option      ......... 0
Bad Circuit ID            ......... 0
Missing Circuit ID        ......... 0
Bad Remote ID             ......... 0
Missing Remote ID         ......... 0
Option Insert Failed      ......... 0
DHCPv6 Requests In        ......... 0
DHCPv6 Replies In         ......... 0
DHCPv6 Relayed to Server  ......... 0
DHCPv6 Relayed to Client  ......... 0
```

Table 90-2: Parameters in the output of the show counter dhcp-relay command

Parameter	Description
`Requests In`	The number of DHCP Request messages received from clients.
`Replies In`	The number of DHCP Reply messages received from servers.
`Relayed To Server`	The number of DHCP Request messages relayed to servers.
`Relayed To Client`	The number of DHCP Reply messages relayed to clients.
`Out To Server Failed`	The number of failures when attempting to send request messages to servers. This is an internal debugging counter.
`Out To Client Failed`	The number of failures when attempting to send reply messages to clients. This is an internal debugging counter.
`Invalid hlen`	The number of incoming messages dropped due to an invalid hlen field.

Table 90-2: Parameters in the output of the show counter dhcp-relay

Parameter	Description
`Bogus giaddr`	The number of incoming DHCP Reply messages dropped due to the bogus giaddr field.
`Corrupt Agent Option`	The number of incoming DHCP Reply messages dropped due to a corrupt relay agent information option field. Note that Agent Option counters only increment on errors occurring if the **ip dhcp-relay agent-option** command is configured for an interface. Messages generating the errors are only dropped if the **ip dhcp-relay agent-option checking** command is configured on the interface as well as the **ip dhcp-relay agent-option** command.
`Missing Agent Option`	The number of incoming DHCP Reply messages dropped due to a missing relay agent information option field. Note that Agent Option counters only increment on errors occurring if the **ip dhcp-relay agent-option** command is configured for an interface. Messages generating the errors are only dropped if the **ip dhcp-relay agent-option checking** command is configured on the interface as well as the **ip dhcp-relay agent-option** command.
`Bad Circuit ID`	The number of incoming DHCP Reply messages dropped due to a bad circuit ID. Note that Agent Option counters only increment on errors occurring if the **ip dhcp-relay agent-option** command is configured for an interface. Messages generating the errors are only dropped if the **ip dhcp-relay agent-option checking** command is configured on the interface as well as the **ip dhcp-relay agent-option** command
`Missing Circuit ID`	The number of incoming DHCP Reply messages dropped due to a missing circuit ID. Note that Agent Option counters only increment on errors occurring if the **ip dhcp-relay agent-option** command is configured for an interface. Messages generating the errors are only dropped if the **ip dhcp-relay agent-option checking** command is configured on the interface as well as the **ip dhcp-relay agent-option** command
`Bad Remote ID`	The number of incoming DHCP Reply messages dropped due to a bad remote ID. Note that Agent Option counters only increment on errors occurring if the **ip dhcp-relay agent-option** command is configured for an interface. Messages generating the errors are only dropped if the **ip dhcp-relay agent-option checking** command is configured on the interface as well as the **ip dhcp-relay agent-option** command
`Missing Remote ID`	The number of incoming DHCP Reply messages dropped due to a missing remote ID. Note that Agent Option counters only increment on errors occurring if the **ip dhcp-relay agent-option** command is configured for an interface. Messages generating the errors are only dropped if the **ip dhcp-relay agent-option checking** command is configured on the interface as well as the **ip dhcp-relay agent-option** command

Table 90-2: Parameters in the output of the show counter dhcp-relay

Parameter	Description
`Option Insert Failed`	The number of incoming DHCP Request messages dropped due to an error adding the DHCP Relay Agent information (option-82). This counter increments when: ■ the DHCP Relay Agent is set to drop packets with the DHCP Relay Agent Option 82 field already filled by another DHCP Relay Agent. This policy is set with the **ip dhcp-relay information policy** command. ■ there is a packet error that stops the DHCP Relay Agent from being able to append the packet with its DHCP Relay Agent Information Option (Option 82) field.
`Note that the following parameters are only used on the Global VRF lite instance when DHCPv6 is running`	
`DHCPv6 Requests In`	The number of incoming DHCPv6 Request messages.
`DHCPv6 Replies In`	The number of incoming DHCPv6 Reply messages.
`DHCPv6 Relayed to Server`	The number of DHCPv6 messages relayed to the server.
`DHCPv6 Relayed to Client`	The number of DHCPv6 messages relayed to the client.

show counter dhcp-server

This command shows counters for the DHCP server on your device.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show counter dhcp-server
```

Mode

User Exec and Privileged Exec

Example

To display counters for the DHCP server on your device, use the command:

```
awplus# show counter dhcp-server
```

Output

Figure 90-3: Example output from the show counter dhcp-server command

```
DHCP server counters
DHCPDISCOVER in       ......... 20
DHCPREQUEST in        ......... 12
DHCPDECLINE in        ......... 1
DHCPRELEASE in        ......... 0
DHCPINFORM in         ......... 0
DHCPOFFER out         ......... 8
DHCPACK out           ......... 4
DHCPNAK out           ......... 0
BOOTREQUEST in        ......... 0
BOOTREPLY out         ......... 0
```

Table 90-3: Parameters in the output of the show counter dhcp-server command

Parameter	Description
`DHCPDISCOVER in`	The number of Discover messages received by the DHCP server.
`DHCPREQUEST in`	The number of Request messages received by the DHCP server.
`DHCPDECLINE in`	The number of Decline messages received by the DHCP server.
`DHCPRELEASE in`	The number of Release messages received by the DHCP server.
`DHCPINFORM in`	The number of Inform messages received by the DHCP server.
`DHCPOFFER out`	The number of Offer messages sent by the DHCP server.
`DHCPACK out`	The number of Acknowledgement messages sent by the DHCP server.
`DHCPNAK out`	The number of Negative Acknowledgement messages sent by the DHCP server. The server sends these after receiving a request that it cannot fulfil because either there are no available IP addresses in the related address pool, or the request has come from a client that doesn't fit the network setting for an address pool.

Table 90-3: Parameters in the output of the show counter dhcp-server

Parameter	Description
`BOOTREQUEST in`	The number of bootp messages received by the DHCP server from bootp clients.
`BOOTREPLY out`	The number of bootp messages sent by the DHCP server to bootp clients.

Related Commands

service dhcp-server
show ip dhcp binding
show ip dhcp server statistics
show ip dhcp pool

show dhcp lease

This command shows details about the leases that the DHCP client has acquired from a DHCP server for interfaces on the device.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show dhcp lease [<interface>]`

Parameter	Description
<interface>	Interface name to display DHCP lease details for.

Mode User Exec and Privileged Exec

Example To show the current lease expiry times for all interfaces, use the command:

```
awplus# show dhcp lease
```

To show the current lease for vlan1, use the command:

```
awplus# show dhcp lease vlan1
```

Output **Figure 90-4: Example output from the show dhcp lease command**

```
Interface vlan1
-----------------------------------------------------------------
IP Address:                   192.168.22.4
Expires:                      13 Mar 2007 20:10:19
Renew:                        13 Mar 2007 18:37:06
Rebind:                       13 Mar 2007 19:49:29
Server:
Options:
  subnet-mask                 255.255.255.0
  routers                     19.18.2.100,12.16.2.17
  dhcp-lease-time             3600
  dhcp-message-type           5
  domain-name-servers         192.168.100.50,19.88.200.33
  dhcp-server-identifier      192.168.22.1
  domain-name                 alliedtelesis.com

Interface vlan2
-----------------------------------------------------------------
IP Address:                   100.8.16.4
Expires:                      13 Mar 2007 20:15:39
Renew:                        13 Mar 2007 18:42:25
Rebind:                       13 Mar 2007 19:54:46
Server:
Options:
  subnet-mask                 255.255.0.0
  routers                     10.58.1.51
  dhcp-lease-time             1000
  dhcp-message-type           5
  dhcp-server-identifier      100.8.16.1
```

Related Commands **ip address dhcp**

show ip dhcp binding

This command shows the lease bindings that the DHCP server has allocated clients.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip dhcp binding [<ip-address>|<address-pool>]`

Parameter	Description
<ip-address>	IPv4 address of a leased IP address, in dotted decimal notation. This displays the lease information for the specified IP address.
<address-pool>	Name of an address pool. This displays the lease information for all clients within the address pool.

Mode User Exec and Privileged Exec

Examples To display all leases for every client in all address pools, use the command:

```
awplus# show ip dhcp binding
```

To display the details for the leased IP address `172.16.2.16`, use the command:

```
awplus# show ip dhcp binding 172.16.2.16
```

To display the leases from the address pool `MyPool`, use the command:

```
awplus# show ip dhcp binding MyPool
```

Output **Figure 90-5: Example output from the show ip dhcp binding command**

```
Pool 30_2_network Network 172.16.2.0/24
DHCP Client Entries
IP Address      ClientId                    Type           Expiry
-------------------------------------------------------------------------------
172.16.2.100    0050.fc82.9ede              Dynamic        21 Sep 2007 19:02:58
172.16.2.101    000e.a6ae.7c14              Static         Infinite
172.16.2.102    000e.a6ae.7c4c              Static         Infinite
172.16.2.103    000e.a69a.ac91              Static         Infinite
172.16.2.104    00e0.189d.5e41              Static         Infinite
172.16.2.150    00e0.2b04.5800              Static         Infinite
172.16.2.167    4444.4400.35c3              Dynamic        21 Sep 2007 14:58:41
```

Related Commands
clear ip dhcp binding
ip dhcp pool
lease
range
service dhcp-server
show ip dhcp pool

show ip dhcp pool

This command displays the configuration details and system usage of the DHCP address pools configured on the device.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip dhcp pool [<address-pool>]`

Parameter	Description
<address-pool>	Name of a specific address pool. This displays the configuration of the specified address pool only.

Mode User Exec and Privileged Exec

Example

```
awplus# show ip dhcp pool
```

Output **Figure 90-6: Example output from the show ip dhcp pool command**

```
Pool p1 :
  network: 192.168.1.0/24
  address ranges:
    addr: 192.168.1.10 to 192.168.1.18
  static host addresses:
    addr: 192.168.1.12     MAC addr: 1111.2222.3333
  lease <days:hours:minutes:seconds> <1:0:0:0>
  subnet mask: 255.255.255.0 (pool's network mask)
  Probe:                        Default Values
    Status:       Enabled       [Enabled]
    Type:         ARP           [Ping]
    Packets:      2             [5]
    Timeout:      200 msecs     [200]
  Dynamic addresses:
    Total:        8
    Leased:       2
    Utilization:  25.0 %
  Static host addresses:
    Total:        1
    Leased:       1
```

Output

Figure 90-7: Example output from the show ip dhcp pool command with IP address 192.168.1.12 assigned to a VLAN interface on the device:

```
Pool p1 :
  network: 192.168.1.0/24
  address ranges:
    addr: 192.168.1.10 to 192.168.1.18
            (interface addr 192.168.1.12 excluded)
            (static host addr 192.168.1.12 excluded)
  static host addresses:
    addr: 192.168.1.12     MAC addr: 1111.2222.3333
            (= interface addr, so excluded)
  lease <days:hours:minutes:seconds> <1:0:0:0>
  subnet mask: 255.255.255.0 (pool's network mask)
  Probe:                        Default Values
    Status:       Enabled       [Enabled]
    Type:         ARP           [Ping]
    Packets:      2             [5]
    Timeout:      200 msecs     [200]
  Dynamic addresses:
    Total:        8
    Leased:       2
    Utilization:  25.0 %
  Static host addresses:
    Total:        1
    Leased:       1
```

Table 90-4: Parameters in the output of the show ip dhcp pool command

Parameter	Description
`Pool`	Name of the pool.
`network`	Subnet and mask length of the pool.
`address ranges`	Individual IP addresses and address ranges configured for the pool. The DHCP server can offer clients an IP address from within the specified ranges only. Any of these addresses that match an interface address on the device, or a static host address configured in the pool, will be automatically excluded from the range, and a message to this effect will appear beneath the range entry.
`static host addresses`	The static host addresses configured on the pool. Each IP address is permanently assigned to the client with the matching MAC address. Any of these addresses that match an interface address on the device will be automatically excluded, and a message to this effect will appear beneath the static host entry.
`lease <days:hours:minutes>`	The lease duration for address allocated by this pool.
`domain`	The domain name sent by the pool to clients. This is the domain name that the client should use when resolving host names using DNS.
`subnet mask`	The subnet mask sent by the pool to clients.
`Probe - Status`	Whether lease probing is enabled or disabled.
`Probe - Type`	The lease probe type configured. Either ping or ARP.
`Probe - Packets`	The number of packets sent for each lease probe in the range 0 to 10.

Table 90-4: Parameters in the output of the show ip dhcp pool command(cont.)

Parameter	Description
`Probe - Timeout`	The timeout value in milliseconds to wait for a response after each probe packet is sent. In the range 50 to 5000.
`dns servers`	The DNS server addresses sent to by the pool to clients.
`default-router(s)`	The default router addresses sent by the pool to clients.
`user-defined options`	The list of user-defined options sent by the pool to clients.
`Dynamic addresses - Total`	The total number of IP addresses that have been configured in the pool for dynamic allocation to DHCP clients.
`Dynamic addresses - Leased`	The number of IP addresses in the pool that have been dynamically allocated (leased) to DHCP clients.
`Dynamic addresses - Utilization`	The percentage of IP addresses in the pool that are currently dynamically allocated to clients.
`Static host addresses - Total`	The number of static IP addresses configured in the pool for specific DHCP client hosts.
`Static host addresses - Leased`	The number of static IP addresses assigned to specific DHCP client hosts.

Related Commands

ip dhcp pool
probe enable
probe packets
probe timeout
probe type
range
service dhcp-server
subnet-mask

show ip dhcp-relay

This command shows the configuration of the DHCP Relay Agent on each interface.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ip dhcp-relay [interface <interface-name>]`

Parameter	Description
<interface-name>	Name of a specific interface. This displays the DHCP configuration for the specified interface only.

Mode User Exec and Privileged Exec

Example To display the DHCP Relay Agent's configuration on the interface `vlan100`, use the command:

```
awplus# show ip dhcp-relay interface vlan100
```

Output **Figure 90-8: Example output from the show ip dhcp-relay command**

```
DHCP Relay Service is enabled

vlan100 is up, line protocol is up
Maximum hop count is 10
Insertion of Relay Agent Option is disabled
Checking of Relay Agent Option is disabled
The Remote Id string for Relay Agent Option is 0000.cd28.074c
Relay information policy is to append new relay agent
information
List of servers :   192.168.1.200
```

Related Commands
ip dhcp-relay agent-option
ip dhcp-relay agent-option checking
ip dhcp-relay information policy
ip dhcp-relay maxhops
ip dhcp-relay server-address

show ip dhcp server statistics

This command shows statistics related to the DHCP server.

You can display the server counters using the **show counter dhcp-server** command as well as with this command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip dhcp server statistics
```

Mode

User Exec and Privileged Exec

Example

To display the server statistics, use the command:

```
awplus# show ip dhcp server statistics
```

Output

Figure 90-9: Example output from the show counter dhcp server statistics command

```
DHCP server counters
DHCPDISCOVER in       ......... 20
DHCPREQUEST in        ......... 12
DHCPDECLINE in        ......... 1
DHCPRELEASE in        ......... 0
DHCPINFORM in         ......... 0
DHCPOFFER out         ......... 8
DHCPACK out           ......... 4
DHCPNAK out           ......... 0
BOOTREQUEST in        ......... 0
BOOTREPLY out         ......... 0
DHCPLEASEQUERY in       ....... 0
DHCPLEASEUNKNOWN out    ....... 0
DHCPLEASEACTIVE out     ....... 0
DHCPLEASEUNASSIGNED out ....... 0
```

Figure 90-10: Parameters in the output of the show counter dhcp server statistics command

Parameter	Description
DHCPDISCOVER in	The number of Discover messages received by the DHCP server.
DHCPREQUEST in	The number of Request messages received by the DHCP server.
DHCPDECLINE in	The number of Decline messages received by the DHCP server.
DHCPRELEASE in	The number of Release messages received by the DHCP server.
DHCPINFORM in	The number of Inform messages received by the DHCP server.
DHCPOFFER out	The number of Offer messages sent by the DHCP server.
DHCPACK out	The number of Acknowledgement messages sent by the DHCP server.

Figure 90-10: Parameters in the output of the show counter dhcp server statistics command(cont.)

Parameter	Description
`DHCPNAK out`	The number of Negative Acknowledgement messages sent by the DHCP server. The server sends these after receiving a request that it cannot fulfil because either there are no available IP addresses in the related address pool, or the request has come from a client that doesn't fit the network setting for an address pool.
`BOOTREQUEST in`	The number of bootp messages received by the DHCP server from bootp clients.
`BOOTREPLY out`	The number of bootp messages sent by the DHCP server to bootp clients.
`DHCPLEASEQUERY in`	The number of Lease Query messages received by the DHCP server from DHCP Relay Agents.
`DHCPLEASEUNKNOWN out`	The number of Lease Unknown messages sent by the DHCP server to DHCP Relay Agents.
`DHCPLEASEACTIVE out`	The number of Lease Active messages sent by the DHCP server to DHCP Relay Agents.
`DHCPLEASEUNASSIGNED out`	The number of Lease Unassigned messages sent by the DHCP server to DHCP Relay Agents.

Related Commands

show counter dhcp-server
service dhcp-server
show ip dhcp binding
show ip dhcp pool

show ip dhcp server summary

This command shows the current configuration of the DHCP server. This includes:

- whether the DHCP server is enabled
- whether the DHCP server is configured to ignore BOOTP requests
- whether the DHCP server is configured to support DHCP lease queries
- the details of any user-defined options
- a list of the names of all DHCP address pools currently configured

This show command does not include any configuration details of the address pools. You can display these using the **show ip dhcp pool** command.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax

```
show ip dhcp server summary
```

Mode

User Exec and Privileged Exec

Example

To display the current configuration of the DHCP server, use the command:

```
awplus# show ip dhcp server summary
```

Output

Figure 90-11: Example output from the show ip dhcp server summary command

```
DHCP Server service is disabled
BOOTP ignore is disabled
DHCP leasequery support is disabled
Pool list: p2
```

Related Commands

ip dhcp leasequery enable
ip dhcp pool
service dhcp-server

subnet-mask

This command sets the subnet mask option for a DHCP address pool you are configuring. Use this command to specify the client's subnet mask as defined in RFC 950. This sets the subnet details using the pre-defined option 1. Note that if you create a user-defined option 1 using the **option** command, then you will override any settings created with this command. If you do not specify a subnet mask using this command, then the pool's network mask (specified using the **next-server** command) is applied.

The **no** variant of this command removes a subnet mask option from a DHCP pool. The pool reverts to using the pool's network mask.

Syntax

```
subnet-mask <mask>

no subnet-mask
```

Parameter	Description
<mask>	Valid IPv4 subnet mask, in dotted decimal notation.

Mode DHCP Configuration

Examples To set the subnet mask option to `255.255.255.0` for DHCP pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# subnet-mask 255.255.255.0
```

To remove the subnet mask option from DHCP pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ip dhcp pool P2
awplus(dhcp-config)# no subnet-mask
```

Related Commands

default-router
dns-server
domain-name
next-server
option
service dhcp-server
show ip dhcp pool

Chapter 91: DHCP for IPv6 (DHCPv6) Introduction and Configuration

DHCPv6 Introduction 91.2
- DHCPv6 for IPv6 ..... 91.3
- DHCPv6 Prefix Delegation 91.3
- DHCPv6 RFCs 91.4
- DHCPv6 Messages 91.5
- DHCPv6 Renewal and Rebinding 91.8
- Stateful DHCPv6 Message Exchange 91.9
- Stateless DHCPv6 Message Exchange 91.10
- DHCPv6 Relay Agent Stateful Message Exchange 91.11
- DHCPv6 Prefix Delegation Message Exchange 91.12

DHCPv6 Client and Server Identification 91.13
DHCPv6 Server and Client Functionality 91.14
- DHCPv6 Server Functionality 91.14
- DHCPv6 Client Functionality 91.15

Configuring DHCPv6 Prefix Delegation 91.17
- Configuring the DHCPv6 Server Delegation Pool 91.17
- Configuring the DHCPv6 PD Client 91.18
- Configuring DHCPv6 Server/Stateful Client (Prefix) 91.19
- Configuring DHCPv6 Server/Stateful Client (Range) 91.20
- Configuring DHCPv6 Server/Stateless Client 91.21
- Configuring DHCPv6 Relay / Server / Client 91.22
- Configuring PD Server / PD Client / Stateless Client 91.27
- Configuring PD via DHCPv6 Relay 91.29
- Configuring PD Subdelegation with SLAAC 91.32
- Configuring PD Subdelegation for Multiple VLANs 91.35
- Configuring DHCPv6 Relay with Recursive PD Subdelegation 91.38

PD Subdelegation System Configuration 91.42
- Stateful_Client Configuration 91.44
- Stateless_Client Configuration 91.44
- PD_Client Configuration 91.44
- DHCPv6_Relay Configuration 91.45
- PD_Subdelegation Configuration 91.46
- PD_Server1 Configuration 91.47
- PD_Server2 Configuration 91.47

DHCPv6 Introduction

This chapter describes the Dynamic Host Configuration Protocol for IPv6 (DHCPv6), including Prefix Delegation (PD) support, provided by your switch. It includes sample configurations to configure your switch as a DHCPv6 Server to pass configuration information to IPv6 clients, and for configuring the switch as a DHCPv6 Client, to request IPv6 addresses from a DHCPv6 Server. This chapter also includes sample configurations to configure your switch for DHCPv6 Prefix Delegation and DHCPv6 Relay Agent features.

DHCPv6 is specified in RFC 3315 "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)". IPv6 Prefix Delegation is specified in RFC 3769 "Requirements for IPv6 Prefix Delegation". DHCPv6 Prefix Delegation is specified in RFC 3633 "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6".

See the section **DHCPv6 Messages** for the sequence of messages sent and received between DHCPv6 Servers and DHCPv6 Clients and brief descriptions of the messages.

For the syntaxes, parameters, descriptions, defaults, and examples for all of the commands used in sample DHCPv6 configurations, refer to **Chapter 92, DHCP for IPv6 (DHCPv6) Commands** for the DHCPv6 Client Server PD commands and **Chapter 90, Dynamic Host Configuration Protocol (DHCP) Commands** for the DHCPv6 Relay commands.

Note

The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

DHCPv6 for IPv6

DHCPv6 is used to delegate IPv6 prefixes and to allocate IPv6 addresses. It offers stateful address autoconfiguration, and complements Stateless Address Autoconfiguration (SLAAC) described in RFC 2462, IPv6 Stateless Address Autoconfiguration. Stateless Address Autoconfiguration (SLAAC) allows an IPv6-aware device to be plugged into a network, and given an IPv6 address prefix without manual configuration.

- DHCPv6 Clients first use RA messages to auto-configure themselves with any default IPv6 route(s) via gateway router(s).
- A DHCPv6 Client can then initiate DHCPv6 Prefix Delegation by including an IAPD (Identity Association for Prefix Delegation) option along with an ID in its solicit messages.
- Interface addresses can also be automatically configured that are derived from delegated prefix information.
- A DHCPv6 Server can be configured to set optional M and/or O flags in Router Advertisement (RA) messages that its sends.
- A DHCPv6 Server can be configured to notify clients that they can use a stateful address configuration protocol (for example, DHCPv6 IANA) to obtain an address by setting the single bit Managed Address Configuration (M) flag to 1 in RA messages
- A DHCPv6 Server can be configured to notify clients that they can use a stateful address configuration protocol (for example, DHCPv6) to obtain non-address configuration information such as optional DNS information plus SNTP information by setting the single bit Other Stateful Configuration (O) flag to 1 in RA messages.
- IAPD or IANA delegation pools are configured in a DHCPv6 PD (Prefix Delegation) Server. Prefixes or addresses to be allocated are stored in theses delegation pools.

See the section **"DHCPv6 Client and Server Identification" on page 91.13** for IAPD information. See the section **"DHCPv6 Messages" on page 91.5** for solicit message information.

When the switch is configured as a DHCPv6 Server, it can:

- **delegate prefixes to IPv6 subnets**. Prefixes allow subnets to be selected, rather than a single node. Like IPv4 addresses, a proportion of the left most bits of the address can be used to indicate the subnet (e.g. using slash notation 2001:0db8:1234::/64). EUI64 suffix information can then be appended to form an IPv6 address on an interface. See also the **IPv6 EUI-64 Addressing** section in **Chapter 30, IPv6 Introduction** for further EUI-64 implementation information.
- **assign normal and temporary IPv6 addresses to devices.** An IPv6 address is a hexadecimal string, made up from eight pairs of octets separated by colons, for example 3ffe:2::0:1. Note that these pairs of octets are called segments in IPv6, and there are eight segments in an IPv6 address. Normal addresses are renewed by the server for as long as the device requires an address. Temporary addresses are assigned for a limited time (lease time) and are usually allocated for privacy reasons, as outlined in RFC 3041 "Privacy Extensions for Stateless Address Autoconfiguration in IPv6".

DHCPv6 Prefix Delegation

DHCPv6 Prefix Delegation automates IPv6 prefix assignment in an IPv6 network, and enables user devices to automatically append their own device component for the user device portion of the IPv6 address.

This gives you the ability to centrally control IPv6 addressing used in remote sites where the IPv6 prefix information is the equivalent of the network portion of an IPv4 address.

DHCPv6 Relay is often used with between PE (Provider Edge) devices and CPE (Customer Premises Equipment) devices with DHCPv6 Prefix Delegation. Once a provider delegates prefixes to its customers, the customers can then subnet and assign the prefixes to their internal links.

Providers use the DHCPv6 feature to manage site addressing, subnet, and link updates.

- Stateless DHCPv6 does not require a DHCPv6 Server to maintain any dynamic state for clients, such as Domain Name System (DNS) server addresses. Note that only configuration information is exchanged with stateless prefix delegation.
- Stateful DHCPv6 uses a DHCPv6 Server to centrally manage IPv6 address and prefix assignment. DHCPv6 Clients get IPv6 address or prefix information from the DHCPv6 Server. DHCPv6 Clients can obtain configuration information that is not available from other protocols, such as DNS.

DHCPv6 RFCs

See the below list of DHCPv6 related RFCs:

- RFC 3315 Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

  Stateful DHCPv6 - DHCPv6 has been standardized by the IETF through RFC 3315.
- RFC 3633 IPv6 Prefix Options for DHCPv6
- DHCPv6 Prefix Delegation (DHCPv6-PD) is an extension to DHCPv6 specified in RFC 3315.
- RFC 6276 DHCPv6 Prefix Delegation for Network Mobility (NEMO)

  Specifies DHCPv6 Prefix Delegation use with a mobile router for network mobility.
- RFC 3646 DNS Configuration Options for DHCPv6.
- RFC 3736 Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6 Stateless DHCPv6 is a combination of IPv6 Stateless Address Autoconfiguration (RFC 4862) and Dynamic Host Control Protocol for IPv6 (RFC 3315). It is a counterpart to IPv6 Stateless Address Autoconfiguration, and can be used with or without IPv6 Stateless Address Autoconfiguration to obtain configuration parameters.
- RFC 4649 Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay Agent Remote-ID Option.
- RFC 4841 Neighbor Discovery for IP version 6 (IPv6).
- RFC 4862 IPv6 Stateless Address Autoconfiguration.
- RFC 5007 DHCPv6 Leasequery.

DHCPv6 Messages

DHCPv6 uses multicast and unicast addresses for communication. A multicast address provides the equivalent functionality to an IPv4 broadcast address. It identifies a group of interfaces, and packets are sent to all interfaces in that group. Addresses reserved for DHCPv6 messages are:

- ff02::1:2. This link-scope multicast address is used by clients to communicate with DHCPv6 servers. When the DHCPv6 module is enabled the switch listens to this address.
- ff05::1:3. This site-scope multicast address is used by clients to communicate with DHCPv6 servers. When the DHCPv6 module is enabled the switch listens to this address.

A normal DHCPv6 message exchange involves the following messages:

1. **Solicit** - sent by a DHCPv6 Client to locate DHCPv6 Servers.
2. **Advertise** - sent by a DHCPv6 server to a DHCPv6 Client in answer to the solicit message as an affirmative message that DHCPv6 Server services are available to a DHCPv6 Client.
3. **Request** - sent by a DHCPv6 Client to a DHCPv6 Server to request configuration parameters.
4. **Reply** - sent by a DHCPv6 Server to a DHCPv6 Client with configuration information.
5. **Renew** - sent by a DHCPv6 Client to a DHCPv6 Server requesting an extension to the address lifetime.

An IPv6 address may be assigned to the DHCPv6 Client for a limited or unlimited time. If the address lifetime is limited, it has a preferred lifetime and a (generally longer) valid lifetime.

Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires.

Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.

An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.

Once half the time between address assignment and the preferred lifetime has passed (the T1 time), the client sends a **Renew** message to the server, requesting an extension to the address lifetime.

If the client has not received a reply after 80% of the valid lifetime has passed (the T2 time), it sends a multicast **Rebind** message to discover another DHCPv6 Server

DHCPv6 Client/Server/Relay Agent messages are exchanged over UDP ports 546 and 547. DHCPv6 Clients listen for DHCPv6 messages on UDP port 546, while DHCPv6 Servers and DHCPv6 Relay Agents listen for DHCPv6 messages on UDP port 547.

In a DHCPv6 Client/DHCPv6 Server environment, DHCPv6 messages are sent from the DHCPv6 Clients to DHCPv6 Server destination UDP port 547. DHCPv6 Servers respond to DHCPv6 Client destination UDP port 546.

In a DHCPv6 Client/DHCPv6 Relay Agent/DHCPv6 Server environment, DHCPv6 messages are sent from DHCPv6 Clients to DHCPv6 Relay Agent and DHCPv6 Server destination UDP port 547. DHCPv6 Servers respond to DHCPv6 Relay Agents via destination UDP port 547. DHCPv6 Relay Agents respond to DHCPv6 Client destination UDP port 546.

DHCPv6 message types

See the below list of DHCPv6 Message Types supported in AlliedWare Plus:

- **SOLICIT**:

  A DHCPv6 Client sends a **SOLICIT** message to locate DHCPv6 Servers.

- **ADVERTISE**:

  A DHCPv6 Server sends an **ADVERTISE** message to indicate that it is available for DHCPv6 service in response to a SOLICIT message received from a DHCPv6 Client.

- **REQUEST**:

  A DHCPv6 Client sends a **REQUEST** message to request configuration parameters, including IPv6 addresses.

- **CONFIRM**:

  A DHCPv6 Client sends a **CONFIRM** message to any available DHCPv6 Server to determine whether the IPv6 addresses it was assigned is still appropriate to the link to which the DHCPv6 Client is connected. This can happen when the DHCPv6 Client detects either a link-layer connectivity change, or if it is powered on and one or more leases are still valid. The **CONFIRM** message is used to confirm whether the DHCPv6 Client is still on the same link or whether it has been moved. The actual lease(s) are not validated; just the prefix portion of the delegated IPv6 addresses or IPv6 prefixes.

- **RENEW**:

  A DHCPv6 Client sends a **RENEW** message to the DHCPv6 Server that originally provided the DHCPv6 Client's addresses and configuration parameters, to extend the lifetimes on the IPv6 addresses assigned to the DHCPv6 Client and to update other configuration parameters.

- **REBIND**:

  A DHCPv6 Client sends a **REBIND** message to any available DHCPv6 Server to extend the lifetimes of the IPv6 addresses assigned to the DHCPv6 Client, and to update other configuration parameters. This message is sent after a DHCPv6 Client receives no response to a **RENEW** message.

- **REPLY**:

  A DHCPv6 Server sends a **REPLY** message containing assigned IPv6 addresses and configuration parameters in response to a **SOLICIT**, **REQUEST**, **RENEW**, and **REBIND** message received from a DHCPv6 Client. A DHCPv6 Server sends a **REPLY** message containing configuration parameters in response to an **INFORMATION-REQUEST** message. A DHCPv6 Server sends a **REPLY** message in response to a **CONFIRM** message confirming or denying that the IPv6 addresses assigned to the DHCPv6 Client are appropriate to the link to which the DHCPv6 Client is connected. A DHCPv6 Server sends a **REPLY** message to acknowledge receipt of a **RELEASE** or **DECLINE** message.

- **RELEASE**:

  A DHCPv6 Client sends a **RELEASE** message to the DHCPv6 Server to inform it that the DHCPv6 Client will no longer use one or more of the assigned IPv6 addresses.

- **DECLINE**:

  A DHCPv6 Client sends a **DECLINE** message to a DHCPv6 Server to indicate that the DHCPv6 Client has determined that one or more IPv6 addresses assigned by the

DHCPv6 Server are already in use on the link to which the DHCPv6 Client is connected.

- **RECONFIGURE**:

  A DHCPv6 Server sends a **RECONFIGURE** message to a DHCPv6 Client to inform the DHCPv6 Client that the DHCPv6 Server has new or updated configuration parameters. The DHCPv6 Client is to initiate a **RENEW**/**REPLY** or an **INFORMATION-REQUEST**/ **REPLY** transaction with the DHCPv6 Server to receive updated information.

- **INFORMATION-REQUEST**:

  A DHCPv6 Client sends an **INFORMATION-REQUEST** message to a DHCPv6 Server to request configuration parameters without the assignment of any IPv6 addresses to the DHCPv6 Client.

- **RELAY-FORW**:

  A DHCPv6 Relay Agent sends a **RELAY-FORW** message to relay messages to DHCPv6 Servers, either directly or through another DHCPv6 Relay Agent. The received message, either a DHCPv6 Client message or a **RELAY-FORW** message from another DHCPv6 Relay Agent, is encapsulated in an option in the **RELAY-FORW** message.

- **RELAY-REPL**:

  A DHCPv6 Server sends a **RELAY-REPL** message to a DHCPv6 Relay Agent containing a message that the DHCPv6 Relay Agent delivers to a DHCPv6 Client. The **RELAY-REPL** message may be relayed by other DHCPv6 Relay Agents for delivery to the destination DHCPv6 Relay Agent. The DHCPv6 Server encapsulates the DHCPv6 Client message as an option in the **RELAY-REPL** message, which the DHCPv6 Relay Agent extracts and relays to the DHCPv6 Client.

DHCPv6 client to DHCPv6 server messages

See the below list of DHCPv6 Messages sent from a DHCPv6 Client to a DHCPv6 Server:

- **SOLICIT**
- **REQUEST**
- **CONFIRM**
- **RENEW**
- **REBIND**
- **RELEASE**
- **DECLINE**
- **INFORMATION-REQUEST**

DHCPv6 server to DHCPv6 client messages

See the below list of DHCPv6 Messages sent from a DHCPv6 Server to a DHCPv6 Client:

- **ADVERTISE**
- **REPLY**
- **RECONFIGURE**

DHCPv6 relay to DHCPv6 relay/server messages

See the below DHCPv6 Message sent from a DHCPv6 Relay to a DHCPv6 Relay/Server:

- **RELAY-FORW**

DHCPv6 relay/server to DHCPv6 relay messages

See the below DHCPv6 Message sent from a DHCPv6 Relay/Server to a DHCPv6 Relay:

- **RELAY-REPL**

DHCPv6 Renewal and Rebinding

See the below list of descriptions for DHCPv6 Renewal and Rebinding terminology:

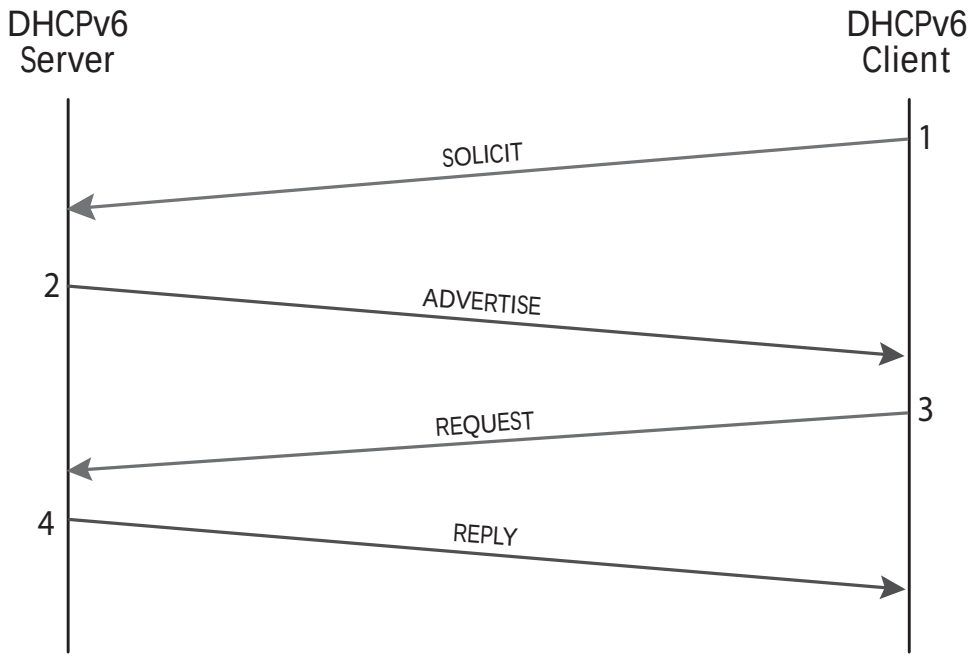
- **T1 Timer:**
  Indicates when a DHCPv6 Client must attempt to renew IPv6 addresses or prefixes.
- **T2 Timer:**
  Indicates when a DHCPv6 Client must attempt to rebind IPv6 addresses or prefixes.
- **Preferred Lifetime:**
  Indicates when preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires. Deprecated addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.
- **Valid Lifetime:**
  Indicates when IPv6 addresses or prefixes must be abandoned. An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.
- **Client States:**
  - **Bound**: Normal operation.
  - **Renewing**: Renewing lease.
  - **Rebinding**: Occurs when no reply has been received from the DHCPv6 Server.
  - **Bound**: Receive, process, and finalize new lease.
- **Renewal Process:**
  - Renewal T1 Timer expires (the default T1 Timer period is 50% of the lease length)
  - DHCPv6 Client transitions from Bound to Renewing state.
  - DHCPv6 Client sends DHCPv6 Request/Renew messages (**REQUEST**/**RENEW** message types)
  - DHCPv6 Client transitions to Rebinding state (if no reply from DHCPv6 Server and the T2 Timer expires).
  - DHCPv6 Client transitions to Bound state (when DHCPv6 Client receives a reply from DHCPv6 Server).

Stateful DHCPv6 Message Exchange

The sequence for stateful DHCPv6 message exchange between a DHCPv6 Client and a DHCPv6 Server is shown below in **Figure 91-1**.

1. The DHCPv6 Client starts by sending a `SOLICIT` message to the DHCPv6 Server.
2. The DHCPv6 Server sends an `ADVERTISE` message back to the DHCPv6 Client.
3. The DHCPv6 Client sends a `REQUEST` message to the DHCPv6 Server.
4. The DHCPv6 Server sends a `REPLY` message back to the DHCPv6 Client to finish.

Figure 91-1: Stateful DHCPv6 Message Exchange Diagram



See section **"DHCPv6 Messages" on page 91.5** for descriptions of all DHCPv6 messages.

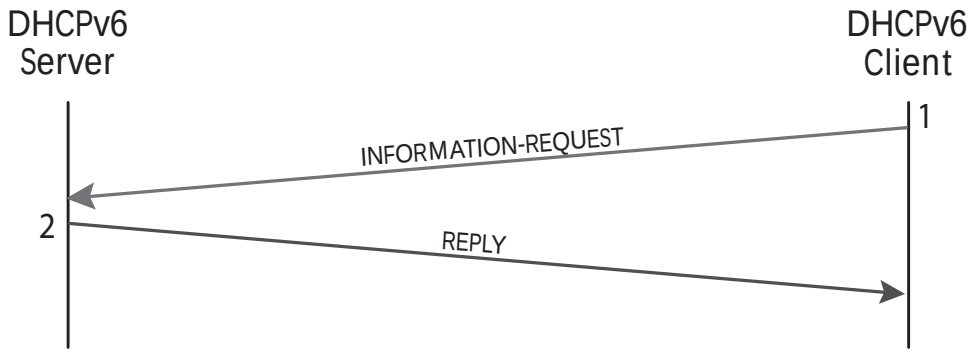
Stateless DHCPv6 Message Exchange

The sequence for stateless DHCPv6 message exchange between a DHCPv6 Client and a DHCPv6 Server is shown below in **Figure 91-2**.

1. The DHCPv6 Client starts by sending an `INFORMATION-REQUEST` message to the DHCPv6 Server. This request specifically excludes the assignment of any IPv6 address.

2. The DHCPv6 Server sends a `REPLY` message back to the DHCPv6 Client to finish.

Figure 91-2: Stateless DHCPv6 Message Exchange Diagram



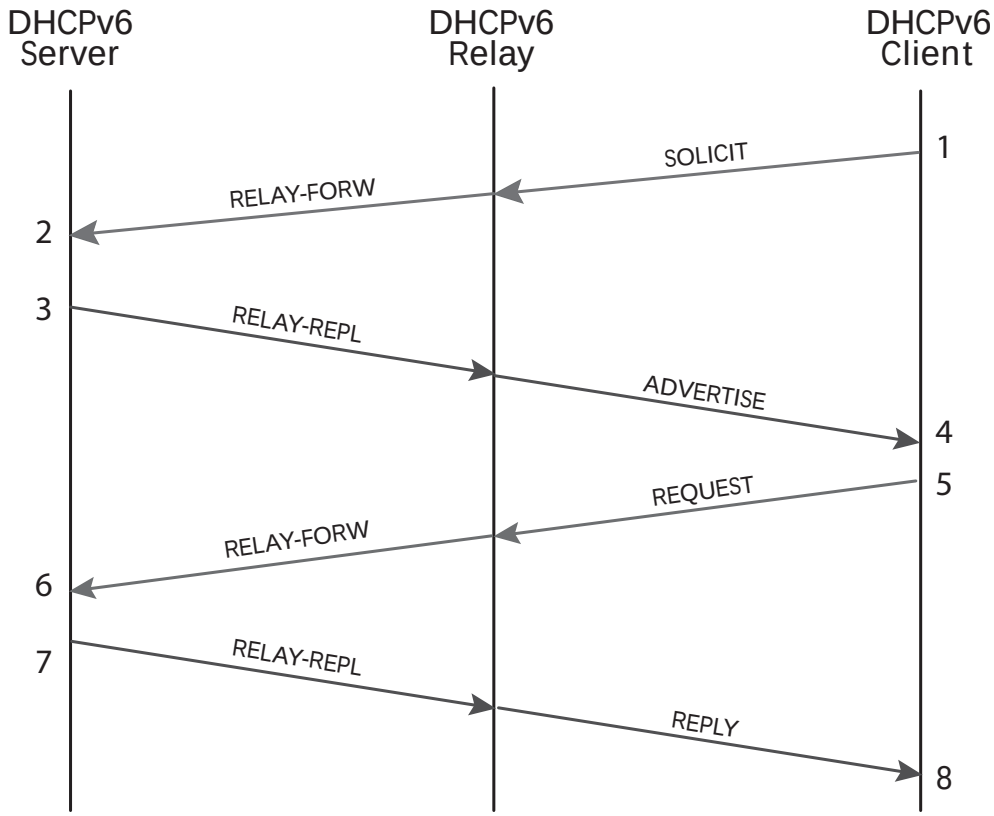
See section **"DHCPv6 Messages" on page 91.5** for descriptions of all DHCPv6 messages.

DHCPv6 Relay Agent Stateful Message Exchange

The sequence for a stateful DHCPv6 message exchange between a DHCPv6 Client, a DHCPv6 Relay, and a DHCPv6 Server is shown below in **Figure 91-3**.

1. The DHCPv6 Client starts by sending a `SOLICIT` message to the DHCPv6 Relay.
2. The DHCPv6 Server receives a `RELAY-FORW` message sent from the DHCPv6 Relay.
3. The DHCPv6 Server sends a `RELAY-REPL` message back to the DHCPv6 Relay.
4. The DHCPv6 Client receives an `ADVERTISE` message sent from the DHCPv6 Relay.
5. The DHCPv6 Client sends a `REQUEST` message to the DHCPv6 Relay.
6. The DHCPv6 Server receives a `RELAY-FORW` message sent from the DHCPv6 Relay.
7. The DHCPv6 Server sends a `RELAY-REPL` message back to the DHCPv6 Relay.
8. TheDHCPv6 Client receives a `RELPY` message sent from the DHCPv6 Relay to finish.

Figure 91-3: DHCPv6 Relay Agent Stateful Message Exchange Diagram



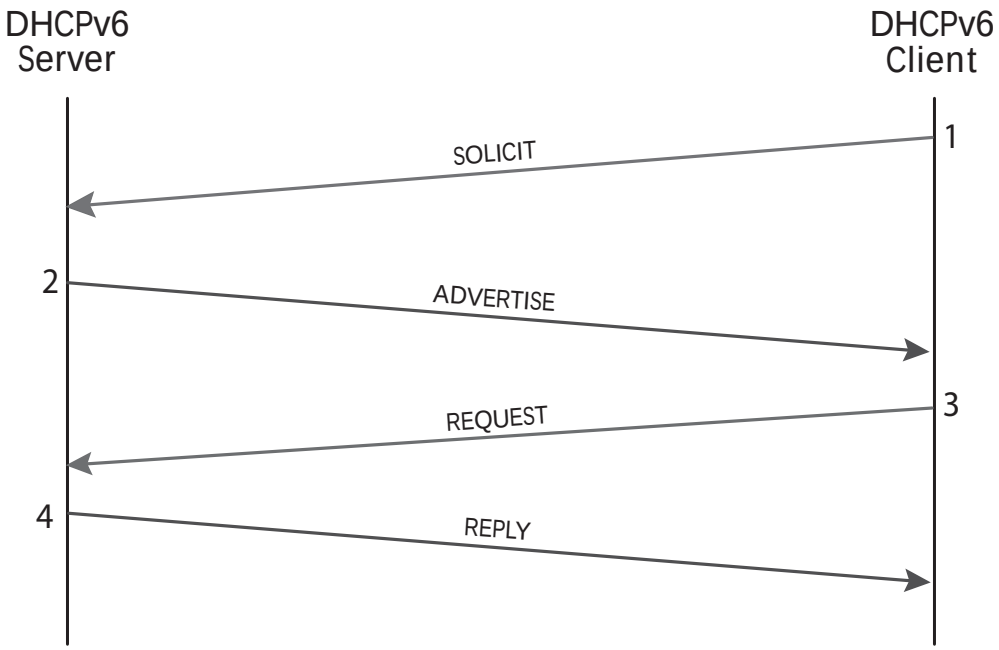
See section **"DHCPv6 Messages" on page 91.5** for message descriptions.

DHCPv6 Prefix Delegation Message Exchange

The sequence for stateful DHCPv6 message exchange between a DHCPv6 Client and a DHCPv6 Server configured for DHCPv6 Prefix Delegation is shown below in **Figure 91-4**.

1. The DHCPv6 Client starts by sending a `SOLICIT` message to the DHCPv6 Server.
2. The DHCPv6 Server sends an `ADVERTISE` message back to the DHCPv6 Client.
3. The DHCPv6 Client sends a `REQUEST` message to the DHCPv6 Server.
4. The DHCPv6 Server sends a `REPLY` message back to the DHCPv6 Client to finish.

Figure 91-4: DHCPv6 Prefix Delegation Message Exchange Diagram



See section **"DHCPv6 Messages" on page 91.5** for descriptions of all DHCPv6 messages.

DHCPv6 Client and Server Identification

DHCPv6 Clients and Servers are identified by a DHCPv6 Unique Identifier (DUID). The DUID consists of a number, followed by a hexadecimal string that identifies the hardware type of the client and the link-layer address of the client.

DHCPv6 Unique Identifier (DUID)

A DUID identifies each DHCPv6 Client and Server, where Client and Server identifier options contain the DUID. The DUID is unique for DHCPv6 Clients and Servers. DHCPv6 uses DUIDs with link-layer addresses for client and server identifiers. The device uses the MAC address from the lowest-numbered interface to determine a DUID.

Prefixes are considered to be for different clients when a DHCPv6 Client requests two prefixes with the same DUID and with different IAIDs on different network interfaces.

Identity associations (IAPD, IAID, and IANA)

DHCPv6 Clients use identity associations to identify each interface that is configured by DHCPv6. An interface's identity association contains the configuration settings of the interface and an Identity Association Identifier (IAID). When the client requests settings from the server for a particular interface, it includes the IAID, to identify the interface.

An Identity Association for Prefix Delegation (IAPD) is a set of IPv6 prefixes assigned to a requesting device. A requesting device may have more than one IAPD where an IAPD is assigned per interface and the device has multiple interfaces.

IAPDs are identified by IAIDs. IAIDs are chosen by requesting devices and are unique among IAPD IAIDs on requesting devices. IAIDs are consistent after reloading using information from the associated interface that is permanently attached to the device.

IAPD, IAID, and IANA use with prefix delegation

For subscriber LAN addressing, you can use DHCPv6 address and/or prefix delegation to provision global IPv6 addresses to subscribers on the LAN.

IAPD or IANA (Identity Association for Non-temporary Addresses) delegation pools are configured in a PD server. Prefixes or addresses to be allocated are stored in delegation pools. RA messages are first used by stateful client(s) to auto-configure themselves with any default IPv6 route(s) via gateway router(s).

When IAPD is specified, a DHCPv6 client can then initiate Prefix Delegation as a requesting device by including an IAPD option with the specified ID in **solicit** messages. Interface addresses can be automatically configured derived from delegated prefix information.

See the section **DHCPv6 Messages** for the sequence of messages sent and received between DHCPv6 Servers and DHCPv6 Clients and brief descriptions of the messages.

DHCPv6 Server and Client Functionality

DHCPv6 Server and Client functionality is mutually exclusive on an interface. When one of these functions is enabled and a user tries to configure a DHCPv6 Client or Server on the same interface, a message is shown to indicate whenever an interface already configured.

DHCPv6 Server Functionality

Configuration parameters for DHCPv6 Clients are configured in DHCPv6 configuration pools. A configuration pool is associated with a particular DHCPv6 Server on an interface. Prefixes that are delegated to DHCPv6 Clients can be specified as a list of assigned IPv6 prefixes for a given DHCPv6 Client or as IPv6 local prefix pools. The list of configured IPv6 prefixes or IPv6 local prefix pools are referenced and used by DHCPv6 configuration pools.

The DHCPv6 Server can provide configuration parameters that do not require the server to maintain any dynamic state for individual clients, such as DNS server addresses and domain search list options. The DHCPv6 Server can be configured for prefix delegation. DHCPv6 Server functionality can be enabled on individual IPv6-enabled VLAN interfaces.

The DHCPv6 Server maintains an automatic binding table in memory to track the assignment of some configuration parameters, such as prefixes between the server and its clients. Automatic bindings can be stored semi-permanently in non-volatile memory.

RA suppression is turned off in the DHCPv6 Server to facilitate neighbor discovery and allow clients via router solicitation to discover gateway routers on the LAN.

DHCPv6 server prefix delegation pool

A DHCPv6 Server Prefix Delegation pool is named and includes information about the configuration parameters that control assignment of prefixes to clients from the pool.

A prefix delegation pool is configured independently and is associated with the DHCPv6 Server by entering the relevant commands at the console. Each DHCPv6 Prefix Delegation pool can contain the following configuration parameters and information:

- SNTP Server IPv6 addresses.
- DNS Server IPv6 addresses.
- DHCPv6 Prefix Delegation information, including a DHCPv6 prefix pool name and available IPv6 prefixes, both with their configured preferred and valid lifetimes.

See the section **Configuring the DHCPv6 Server Delegation Pool** for the sequence of commands issued together with command modes, command syntaxes, and brief descriptions.

IPv6 prefix assignment

A DHCPv6 Server selects prefixes for assignment following a request from a DHCPv6 Client. The DHCPv6 Server selects prefixes for a DHCPv6 Client statically or dynamically. You can statically configure IPv6 prefixes, plus preferred and valid lifetimes for an IAPD, of a DHCPv6 Client as identified by its DUID.

When the delegating DHCPv6 Server receives a request from a DHCPv6 Client, it checks if there is a static binding configured for the IAPD in the message from the DHCPv6 Client. If a static binding is present, prefixes in the binding are returned to the DHCPv6 client. If no binding is found, the DHCPv6 Server can assign prefixes for the DHCPv6 client from DNS.

DHCPv6 Server assigns prefixes dynamically from an IPv6 local prefix pool. When DHCPv6 Server receives a prefix request from a DHCPv6 Client, it attempts to obtain unassigned prefixes from the IPv6 local prefix pool. After the DHCPv6 Client releases the previously assigned IPv6 prefixes, the DHCPv6 server returns them to the pool for reassignment.

Automatic binding table

Each DHCPv6 configuration pool has a linked automatic binding table. The automatic binding table records all IPv6 prefixes in the configuration pool that are delegated to DHCPv6 Clients. Each permanent storage to which the binding database is saved is called the database agent. Database agent binding information is stored in NVRAM so configuration information, including prefixes for DHCPv6 Clients from a DHCPv6 Server, is retained after a reload. Each binding table entry contains the following information:

- DHCPv6 Server network interface for the DHCPv6 configuration pool.
- Configuration pool for the binding table.
- Preferred and valid lifetimes per prefix.
- Prefixes delegated to each IAPD.
- IAPDs associated with the client.
- Client IPv6 address.
- Client DUID.

DHCPv6 Server creates a binding table entry when a prefix is delegated to a DHCPv6 Client from the configuration pool, and the entry is updated when the DHCPv6 Client renews, rebinds, or confirms the prefix delegation.

A binding table entry is deleted when the DHCPv6 Client releases all prefixes in the binding, all valid lifetimes expire, or when you run the **clear ipv6 dhcp binding** command. See the **show ipv6 dhcp binding** command for **clear ipv6 dhcp binding**.

The DHCPv6 Server assign prefixes dynamically from an IPv6 local prefix pool. When the DHCPv6 Server receives a prefix request from a DHCPv6 Client, then the DHCPv6 Server gets unassigned prefixes from the DHCPv6 configuration pool. Once the DHCPv6 client releases assigned prefixes, the DHCPv6 Server returns prefixes to the configuration pool.

SNTP server functionality

The SNTP Server Functionality gives a list of IPv6 addresses for SNTP Servers that the DHCPv6 Client can use to synchronize system time to a standard time server. The DHCPv6 Server lists the SNTP Servers for the DHCPv6 Clients to select from to synchronize timing.

See the **sntp-address** command for further SNTP information and command examples.

DHCPv6 server and DNS for IPv6 address assignment

Additional options, such as the default domain and DNS name-server address, can be passed back to the DHCPv6 Client. Address pools can be assigned for use on a specific interface, or the DHCPv6 Server can automatically find the appropriate pool.

See the **dns-server (DHCPv6)** and **domain-name (DHCPv6)** commands for examples.

DHCPv6 Client Functionality

DHCPv6 Clients can request the delegation of prefixes from a DHCPv6 Server. The IPv6 prefixes acquired from a delegating DHCPv6 Server are stored in an IPv6 prefix pool. The prefixes in the IPv6 prefix pool can be used to number downstream device interfaces.

A DHCPv6 Client is enabled on an individual IPv6-enabled VLAN interface, and can also request and accept those configuration parameters that do not require a DHCPv6 Server to maintain any dynamic state for individual clients, such as DNS server addresses.

See the section **Configuring the DHCPv6 PD Client** for the sequence of commands issued together with command modes, command syntaxes, and brief descriptions.

DHCPv6 server selection by a DHCPv6 client

A DHCPv6 Client builds a list of DHCPv6 servers by sending a **solicit** message and by receiving **advertisement** replies from DHCPv6 Servers. The replies are ranked by the DHCPv6 Client based on preference value, when DHCPv6 Servers add a preference value to their advertisement messages. For DHCPv6 Clients to obtain IPv6 prefixes from DHCPv6 Servers, only DHCPv6 Servers that advertise prefixes are considered by DHCPv6 Clients.

See the section **DHCPv6 Messages** for the sequence of messages sent and received between DHCPv6 Servers and DHCPv6 Clients and brief descriptions of the messages.

Note The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

For the syntaxes, parameters, descriptions, defaults, and examples for all of the commands used in the following DHCPv6 configurations, refer to **Chapter 92, DHCP for IPv6 (DHCPv6) Commands** for the DHCPv6 Client Server PD commands and **Chapter 90, Dynamic Host Configuration Protocol (DHCP) Commands** for the DHCPv6 Relay commands.

See the section **Configuring DHCPv6 Prefix Delegation** for the sequence of commands issued together with command modes, command syntaxes, and brief descriptions.

See the **PD Subdelegation System Configuration** section for a an example network topology and configuration output to adapt for your own network configuration.

Configuring DHCPv6 Prefix Delegation

See the sections listed below when configuring DHCPv6 Prefix Delegation:

- **Configuring the DHCPv6 Server Delegation Pool**
- **Configuring the DHCPv6 PD Client**

Configuring the DHCPv6 Server Delegation Pool

Perform this sequence of command entries to create and configure the DHCPv6 configuration pool and associate it with a DHCPv6 server on a VLAN interface.

Note the links in the list are for the command name in the relevant command chapter, while the links in the table below the list are for the command syntaxes showing mode.

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **ipv6 dhcp pool**
4. **domain-name (DHCPv6)**
5. **dns-server (DHCPv6)**
6. **prefix-delegation pool**
7. **exit**
8. **interface (to configure)**
9. **ipv6 dhcp server**
10. **exit**

Table 91-1: Configuring the DHCPv6 configuration pool

1.	`awplus>enable (Privileged Exec mode)`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)#ipv6 dhcp pool <DHCPv6-poolname>`	Configure a DHCPv6 configuration information pool and enters DHCPv6 configuration mode.
4.	`awplus(config-dhcp6)# domain-name <domain-name>`	Configure a domain name for a DHCPv6 client.
5.	`awplus(config-dhcp6)# dns-server <ipv6-address>`	Specify the DNS IPv6 servers available to a DHCPv6 client.
6.	`awplus(config-dhcp6)# prefix-delegation pool <DHCPv6-poolname> [lifetime {<valid-time>|infinite} {<preferred-time>|infinite}]`	Specify a named IPv6 prefix pool from which prefixes are delegated to DHCPv6 clients.

Table 91-1: Configuring the DHCPv6 configuration pool (cont.)

7.	`awplus(config-dhcp6)# exit`	Exit DHCPv6 configuration mode, and returns the device to Global Configuration mode.
8.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enters Interface Configuration mode.
9.	`awplus(config-if)# ipv6 dhcp-server [<DHCPv6-poolname>]`	Enable DHCPv6 on the specified VLAN interface.
10.	`awplus(config-if)# exit`	Return to Global Configuration mode.

Configuring the DHCPv6 PD Client

General prefixes can be defined dynamically from a prefix received by a DHCPv6 prefix delegation client via IAPD. The delegated prefix is stored in a general prefix:

Commands Applied

1. **enable (Privileged Exec mode)**
2. **configure terminal**
3. **interface (to configure)**
4. **ipv6 dhcp client pd**
5. **exit**

Table 91-2: Configuring the DHCPv6 configuration pool

1.	`awplus>enable (Privileged Exec mode)`	Enter Privileged Exec mode.
2.	`awplus#configure terminal`	Enter Global Configuration mode.
3.	`awplus(config)# interface <interface-list>`	Specify a VLAN interface, and enters Interface Configuration mode.
4.	`awplus(config-if)# ipv6 dhcp client pd <prefix-name>`	Enables the DHCPv6 client process and enables a request for prefix delegation through a specified VLAN interface.
5.	`awplus(config-if)# exit`	Return to Global Configuration mode.

Note that devices can be optionally configured to advertise connected routing information associated with delegated prefixes to other devices within the domain via dynamic routing protocols.

Note the following configuration examples build from a simple two device system up to a complex sub-delegation system involving multiple devices and multiple delegation pools.

Configuring DHCPv6 Server/Stateful Client (Prefix)

In this example, a stateful client can obtain its interface address (via DHCPv6 IANA) and other information (such as DNS, SNTP) configured in the DHCP Server delegation pool.

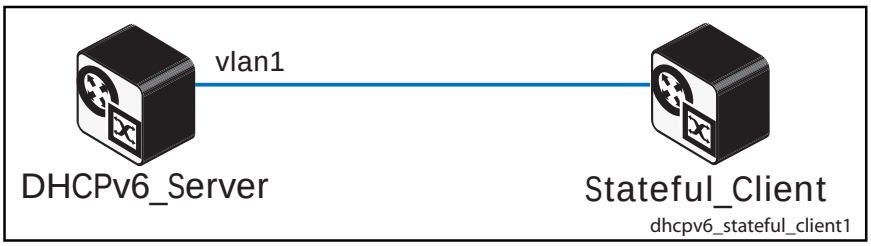
- The client is allocated an address from the address prefix configured in the DHCP Server pool. A stateful client can automatically learn about any gateway router(s) on the LAN and dynamically add associated default route(s) into its routing table via next-hop link-local address of the gateway router via router advertisements.
- RA suppression is turned off in the DHCPv6 Server to facilitate neighbor discovery.

For command information, see the **address prefix** and **ipv6 dhcp server** commands on the DHCPv6 Server and the **ipv6 address dhcp** command on the Stateful Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-5: DHCPv6 Server / Stateful Client configuration topology:



DHCPv6_Server configuration (prefix)

See the below configuration for a device with the hostname **DHCPv6_Server**:

```
hostname DHCPv6_Server
!
ipv6 dhcp pool pool1
 address prefix 2001:db8:10::/64
 dns-server 2001:db8:10::10
 domain-name example.com
 sntp address 2001:db8:10::20
!
interface vlan1
 description to_Stateful_Client
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp server pool1
!
ipv6 forwarding
```

Stateful_Client configuration (IANA)

See the below configuration for a device with the hostname **Stateful_Client**:

```
hostname Stateful_Client
!
interface vlan1
 ipv6 address dhcp
```

Configuring DHCPv6 Server/Stateful Client (Range)

In this example, a stateful client can obtain its interface address (via DHCPv6 IANA) and other information (such as DNS, SNTP) configured in the DHCP Server delegation pool.

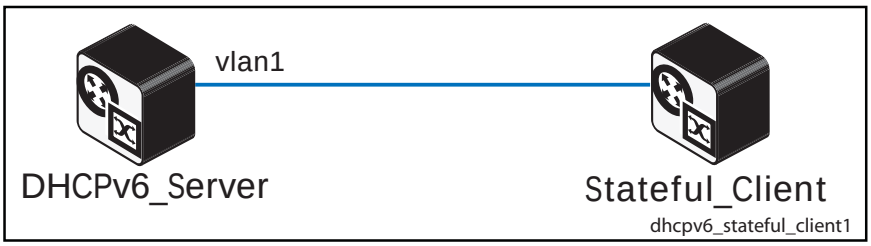
- The client is allocated an address from the address range configured in the DHCP Server pool. A stateful client can automatically learn about any gateway router(s) on the LAN and dynamically add associated default route(s) into its routing table via next-hop link-local address of the gateway router via router advertisements.
- RA suppression is turned off in the DHCPv6 Server to facilitate neighbor discovery.

For command information, see the **address range** and **ipv6 dhcp server** commands on the DHCPv6 Server and the **ipv6 address dhcp** command on the Stateful Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-6: DHCPv6 Server / Stateful Client configuration topology:



DHCPv6_Server configuration (range)

See the below configuration for a device with the hostname **DHCPv6_Server**:

```
hostname DHCPv6_Server
!
ipv6 dhcp pool pool1
 address range 2001:db8:10::2 2001:db8:10::100
 dns-server 2001:db8:10::10
 domain-name example.com
 sntp address 2001:db8:10::20
!
interface vlan1
 description to_Stateful_Client
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp server pool1
!
ipv6 forwarding
```

Stateful_Client configuration (IANA)

See the below configuration for a device with the hostname **Stateful_Client**:

```
hostname Stateful_Client
!
interface vlan1
 ipv6 address dhcp
```

Configuring DHCPv6 Server/Stateless Client

The Stateless Client can automatically learn about any gateway router(s) on the LAN via router solicitation. It can then dynamically add associated default route(s) into its routing table via the next-hop link-local address of the gateway router.

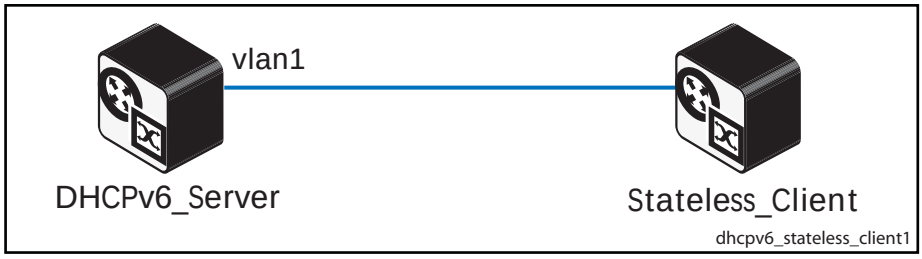
From RFC 4864, section 5.5.3: If the sum of the prefix length and interface identifier length does not equal 128 bits, the Prefix Information option must be ignored.

The effect is that prefix information received in an RA will not be applied to form an IPv6 address via SLAAC if the prefix is greater or less than 64. Since EUI is 64 bits in length the IPv6 Prefix of the advertising device must be 64 bits as well.

For command information, see the **address prefix** and **ipv6 dhcp server** commands on the DHCPv6 Server and the **ipv6 address autoconfig** command on the Stateless Client.

Note **bold** configuration command entries show hostnames and interfaces. Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-7: DHCPv6 Server / Stateless Client configuration topology:



DHCPv6_Server configuration (prefix)

See the below alternative configuration for a device with the hostname **DHCPv6_Server**:

```
hostname DHCPv6_Server
!
ipv6 dhcp pool pool1
 address prefix 2001:db8:10::/64
 dns-server 2001:db8:10::10
 domain-name example.com
 sntp address 2001:db8:10::20
!
interface vlan1
 description to_Stateless_Client
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp server pool1
!
ipv6 forwarding
```

Stateless_Client configuration (SLAAC)

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client
!
interface vlan1
 ipv6 address autoconfig
```

Configuring DHCPv6 Relay / Server / Client

In this example, a stateful client device (via IANA) obtains its interface address from a DHCPv6 Server pool. In this example, the DHCPv6 Client resides in a network that is remote from the DHCPv6 Server and communicates via an intermediate DHCPv6 Relay. Diagnostics command output is shown following the configuration output.

The DHCPv6 Server needs a route to the remote LAN, where the client request originates from via the appropriate next-hop. In this example, the next-hop address is the link-local address of the DHCPv6 Relay. The route could be statically configured as in this example, or dynamically learned via an IPv6 routing protocol.

Link addresses are configured in DHCPv6 Server address pools when there are remote clients that communicate via intermediate relay(s).

When a DHCPv6 Relay receives a request from the DHCPv6 Client, it sends a **relay-forward** message toward the DHCPv6 Server. The DHCPv6 Relay message includes an address from the client-facing interface in the link-address field of the **relay-forward** message. The address of the server-facing interface is used as the IPv6 source of the relay message, and the server will send any reply to that address.

The DHCPv6 Relay automatically adds a route for the delegated prefix into its routing table for the duration of the valid lease, via the interface where the PD client device resides.

When an address on the incoming interface of the DHCPv6 Server or a link address set in the incoming delegation request packet matches the link-address prefix configured in the delegation pool, the server is able to match and use the appropriate delegation pool for relayed delegation request messages.

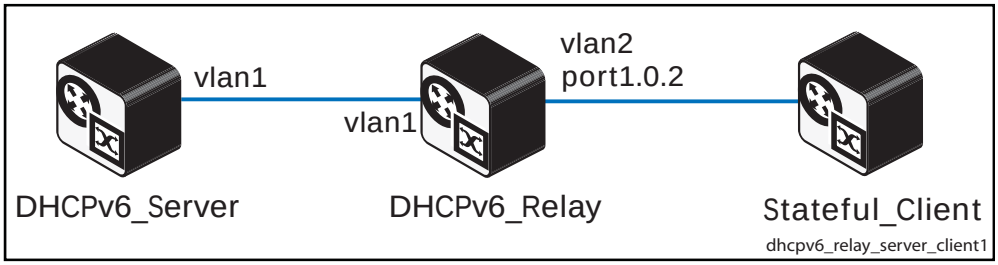
Active bindings are stored in non-volatile memory, and are retained over a device reboot.

For more command information, see the **link-address** and **ipv6 dhcp pool** commands on the DHCPv6 Server, the **ip dhcp-relay server-address** command on the DHCPv6 Relay, and the **ipv6 address dhcp** command on the Stateful Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-8: DHCPv6 server / relay / stateful client configuration topology



DHCPv6_Server configuration

See the below configuration for a device with the hostname **DHCPv6_Server**:

```
hostname DHCPv6_Server
!
ipv6 dhcp pool pool1
 address prefix 2001:db8:20::/64
 link-address 2001:db8:20::/64
 dns-server 2001:db8:10::10
 domain-name example.com
 sntp address 2001:db8:10::20
!
interface vlan1
 description to_DHCPv6_Relay
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
ipv6 route 2001:db8:20::/64 fe80::200:cdff:fe29:a65f vlan1
```

DHCPv6_Server diagnostic output

See the DHCPv6_Server diagnostic output after entering the command to clear bindings:

```
DHCPv6_Server#clear ipv6 dhcp bind all
DHCPv6_Server#show ipv6 interface
Interface     IPv6-Address                    Status     Protocol
lo            unassigned                      admin up   running
vlan1         2001:db8:10::1/64               admin up   running
              fe80::eecd:6dff:fe5a:b864/64
```

When the DHCPv6 Client connects, see the binding and associated counters as below:

```
DHCPv6_Server#show ipv6 dhcp binding
Pool pool1
 Address 2001:db8:20:0:e2e3:7b54:6d72:28b4
         client IAID 77c973a3, DUID 00010001182323a8001577c973a3
         preferred lifetime 604800, valid lifetime 2592000
         starts  at 30 Oct 2012 23:50:01
         expires at 29 Nov 2012 23:50:01
DHCPv6_Server#show counter ipv6 dhcp-server
DHCPv6 server counters
SOLICIT in               ....... 3
ADVERTISE out            ....... 3
REQUEST in               ....... 2
CONFIRM in               ....... 1
RENEW in                 ....... 0
REBIND in                ....... 0
REPLY out                ....... 2
RELEASE in               ....... 0
DECLINE in               ....... 0
INFORMATION-REQUEST in   ....... 0
```

Stateful_Client configuration

See the below configuration for a device with the hostname **Stateful_Client**:

```
hostname Stateful_Client
!
interface vlan1
 ipv6 address dhcp
```

Stateful_Client diagnostic output

See the Stateful Client device diagnostic output after entering the command to clear bindings:

```
Stateful_Client#clear ipv6 dhcp client vlan1
Stateful_Client#show ipv6 interface
Interface     IPv6-Address                                  Status      Protocol
lo            unassigned                                    admin up    running
vlan1         fe80::215:77ff:fec9:73a3/64                   admin up    running
```

Once the client has an address delegated, see the binding and associated counters below:

```
Stateful_Client#show ipv6 interface
Interface     IPv6-Address                                  Status      Protocol
lo            unassigned                                    admin up    running
vlan1         2001:db8:20:0:e2e3:7b54:6d72:28b4/64          admin up    running
              fe80::215:77ff:fec9:73a3/64
Stateful_Client#show count ipv6 dhcp-client
DHCPv6 client counters
SOLICIT out              ....... 3
ADVERTISE in             ....... 1
REQUEST out              ....... 1
CONFIRM out              ....... 0
RENEW out                ....... 0
REBIND out               ....... 0
REPLY in                 ....... 1
RELEASE out              ....... 0
DECLINE out              ....... 0
INFORMATION-REQUEST out  ....... 0
```

DHCPv6_Relay configuration

See the below configuration for a device with the hostname **DHCPv6_Relay**:

```
hostname DHCPv6_Relay
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan1
 description to_DHCPv6_Server
 ipv6 address 2001:db8:10::2/64
 ipv6 enable
!
interface vlan2
 description to_Stateful_Client
 ipv6 address 2001:db8:20::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag
 ip dhcp-relay server-address 2001:db8:10::1 vlan1
!
ipv6 forwarding
```

DHCPv6_Relay diagnostic output

See the DHCPv6_Relay device diagnostic output after entering the command to clear bindings:

```
DHCPv6_Relay#show ipv6 interface
Interface     IPv6-Address                      Status     Protocol
eth0          unassigned                        admin up   running
lo            unassigned                        admin up   running
vlan1         2001:db8:10::2/64                 admin up   running
              fe80::200:cdff:fe29:a65f/64
vlan2         2001:db8:20::1/64                 admin up   running
              fe80::200:cdff:fe29:a65f/64
DHCPv6_Relay#show counter dhcp-relay
DHCP relay counters
Requests In              ....... 0
Replies In               ....... 0
Relayed To Server        ....... 0
Relayed To Client        ....... 0
Out To Server Failed     ....... 0
Out To Client Failed     ....... 0
Invalid hlen             ....... 0
Bogus giaddr             ....... 0
Corrupt Agent Option     ....... 0
Missing Agent Option     ....... 0
Bad Circuit ID           ....... 0
Missing Circuit ID       ....... 0
Bad Remote ID            ....... 0
Missing Remote ID        ....... 0
Option Insert Failed     ....... 0
DHCPv6 Requests In       ....... 5
DHCPv6 Replies In        ....... 3
DHCPv6 Relayed To Server .......5
DHCPv6 Relayed To Client .......3
```

Use the **show log** command to view additional useful information on the system log.

Real time protocol messages for diagnostic purposes can be viewed via verbose **tcpdump** for traffic traversing a VLAN, for example:

```
awplus#tcpdump -vvvni vlan1
```

Configuring PD Server / PD Client / Stateless Client

In this example, the PD Client is assigned a prefix from the PD Server via IAPD.

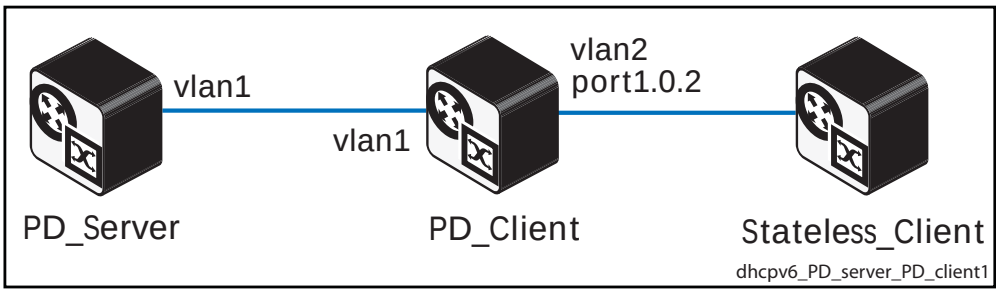
- PD_Client auto-configures an upstream VLAN interface **vlan1** address via SLAAC.
- PD_Client is assigned a prefix from PD_Server via IAPD.
- PD _Client auto-configures its downstream VLAN interface **vlan2** address derived from a combination the delegated prefix and eui64 suffix.
- Stateless_Client auto configures its VLAN interface address using SLAAC.

For more command information, see the **prefix-delegation pool** and **ipv6 dhcp server** commands on the DHCPv6 PD Server, the **ipv6 dhcp client pd** command on the DHCPv6 PD Client, and the **ipv6 address autoconfig** command on the Stateless Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-9: PD_Server /PD_Client / Stateless_Client configuration topology



PD_Server configuration (IAPD)

See the below configuration for a device with the hostname **PD_Server**:

```
hostname PD_server
!
ipv6 local pool pd_vlan1 2001:db8:20::/48 56
!
ipv6 dhcp pool pool1
 prefix-delegation pool pd_vlan1
!
interface vlan1
 description to_PD_Client
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool1
!
ipv6 forwarding
```

Stateless_Client configuration

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client
!
interface vlan1
  ipv6 address autoconfig
```

PD_Client configuration

See the below configuration for a device with the hostname **PD_Client**:

```
hostname PD_Client
!
ipv6 dhcp pool pool1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan1
 description to_PD_Server
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan2
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
ipv6 forwarding
```

Configuring PD via DHCPv6 Relay

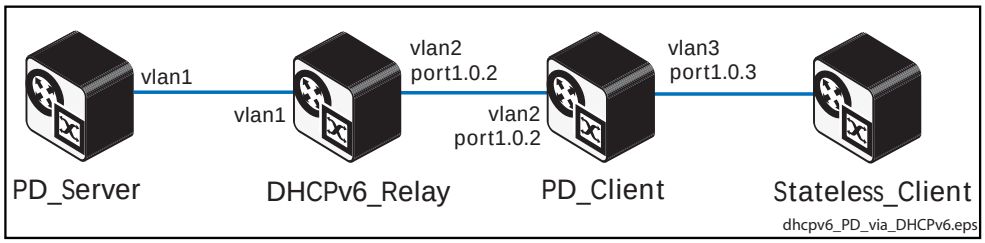
This example includes Prefix Delegation (IAPD) via an intermediate DHCPv6 relay. The DHCPv6 server requires a route to the remote IPv6 subnet of the DHCPv6 relay, where the source of the IA PD request originates from via the link-local address of the DHCPv6 Relay.

For detailed command information, see the **prefix-delegation pool** and **ipv6 dhcp server** commands on the DHCPv6 PD Server, the **ip dhcp-relay server-address** command on the DHCPv6 Relay, the **ipv6 dhcp client pd** command on the DHCPv6 PD Client, and the **ipv6 address autoconfig** command on the Stateless Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-10: PD_Server / DHCPv6_Relay / PD_Client / Stateless_Client configuration topology



PD_Server configuration

See the below configuration for a device with the hostname **PD_Server**:

```
hostname PD_Server
!
ipv6 local pool pd_vlan1_relay 2001:db8:20::/48 56
!
ipv6 dhcp pool pool1
 link-address 2001:db8:20::/64
 prefix-delegation pool pd_vlan1_relay
!
interface vlan1
 description to_DHCPv6_Relay
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
ipv6 route 2001:db8:20::/64 fe80::200:cdff:fe29:a65f vlan1
```

DHCPv6_Relay configuration

See the below configuration for a device with the hostname **DHCPv6_Relay**:

```
hostname DHCPv6_Relay
!
vlan database
 vlan 2 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan1
 description to_PD_Server
 ipv6 address 2001:db8:10::2/64
 ipv6 enable
!
interface vlan2
 description to_PD_Client
 ipv6 address 2001:db8:20::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ip dhcp-relay server-address 2001:db8:10::1 vlan1
!
ipv6 forwarding
```

PD_Client configuration

See the below configuration for a device with the hostname **PD_Client**:

```
hostname PD_Client
!
ipv6 dhcp pool pool1
!
vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan2
 description to_DHCPv6_Relay
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan3
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
ipv6 forwarding
!
```

Stateless_Client configuration

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client
!
interface vlan1
 ipv6 address autoconfig
!
```

Configuring PD Subdelegation with SLAAC

In this example, the PD subdelegation device acts as both a PD client on its upstream interface facing the PD server, and also recursively acts as a PD server, to sub-delegate prefixes to a downstream PD client. The PD subdelegation device also supports router solicitation on downstream interfaces to allow stateless client devices to auto configure themselves via SLAAC (Stateless Address Auto Configuration).

For this example, a longer prefix mask is used for each subdelegated prefix.

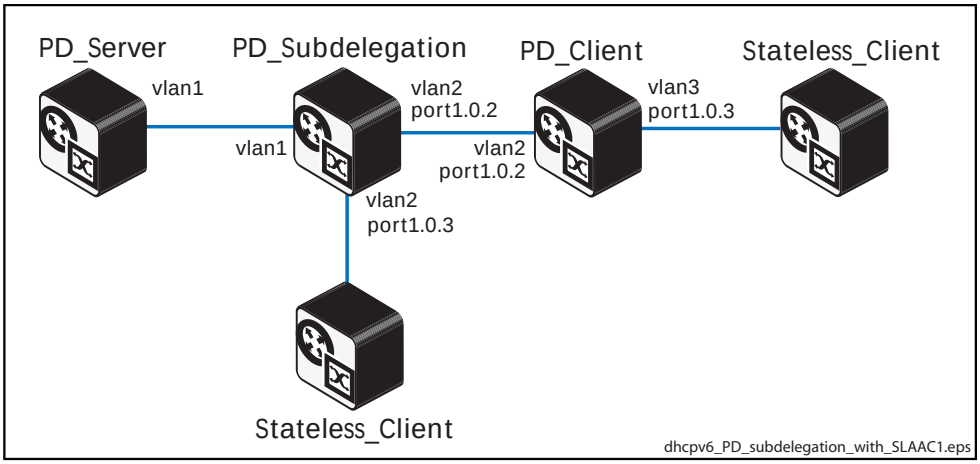
- The PD subdelegation device auto configures its upstream interface facing the PD server via SLAAC.
- The PD server device advertises its prefix to the PD subdelegation device. This is stored in the local address pool in the PD subdelegation device.
- The PD subdelegation device then auto configures its downstream interface based on a combination of the advertised prefix and EUI64 suffix, and advertises Router Advertisements (RAs).
- The PD subdelegation device then recursively advertises prefixes stored in its local address pool out its downstream interface to the PD client device.
- The PD client device uses SLAAC to configure its upstream **vlan2** interface. The PD client device configures its downstream **vlan3** interface based on recursively delegated prefix, and advertises RAs via **vlan3** to the attached Stateless_Client device.
- Stateless_Client devices auto configure their address and gateway router information, using SLAAC, via router solicitation and router advertisements.

For detailed command information, see the **prefix-delegation pool** and **ipv6 dhcp server** commands on the DHCPv6 PD Server and the DHCPv6 PD Subdelegation Server, the **ipv6 dhcp client pd** command on the DHCPv6 PD Client, and the **ipv6 address autoconfig** command on the Stateless Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-11: PD Server / Subdelegation / Client / Stateless Client configuration topology:



PD_Server configuration

See the below configuration for a device with the hostname **PD_Server**:

```
hostname PD_Server
!
ipv6 local pool pd_vlan1 2001:db8:20::/48 56
!
ipv6 dhcp pool pool1
 prefix-delegation pool pd_vlan1
!
interface vlan1
 description to_PD_Subdelegation
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
```

PD_Subdelegation configuration

See the below configuration for a device with the hostname **PD_Subdelegation**:

```
hostname PD_Subdelegation
!
ipv6 local pool pdpool1 pool1 ::/56 64
!
ipv6 dhcp pool pool1
 prefix-delegation pool pdpool1
!
vlan database
 vlan 2 state enable
!
interface port1.0.2-port1.0.3
 switchport
 switchport mode access
 switchport access vlan 2
!
interface vlan1
 description to_PD_Server
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan2
 description to_PD_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
```

PD_Subdelegation diagnostic output

See the below **PD_Subdelegation** device diagnostic output to validate PD configuration:

```
PD_Subdelegation#show ipv6 dhcp interface
vlan1 is in client (Prefix-Delegation) mode
 Prefix name pool1
        prefix 2001:db8:20:fe00::/56
        preferred lifetime 604800, valid lifetime 2592000
        starts  at  8 Nov 2012 14:23:12
        expires at  8 Dec 2012 14:23:12
vlan2 is in server mode
 Using pool  : pool1
 Preference  : 0
```

Stateless_Client configuration

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client
!
interface vlan1
 ipv6 address autoconfig
```

PD_Client configuration

See the below configuration for a device with the hostname **PD_Client**:

```
hostname PD_Client
!
ipv6 dhcp pool pool1
!
vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan2
 description to_PD_Subdelegation
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan3
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
ipv6 forwarding
```

Configuring PD Subdelegation for Multiple VLANs

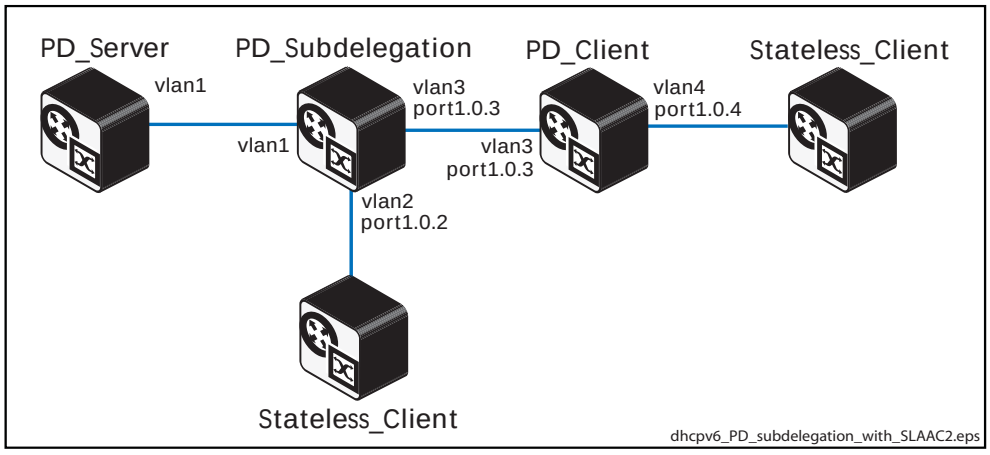
This example is similar to the previous example, **Configuring PD Subdelegation with SLAAC**, with the difference being that the PD subdelegation device has two downstream VLANs (**vlan2** and **vlan3**) instead of one downstream VLAN (**vlan2**) as shown previously.

For detailed command information, see the **prefix-delegation pool** and **ipv6 dhcp server** commands on the DHCPv6 PD Server and the DHCPv6 PD Subdelegation Server, the **ipv6 dhcp client pd** command on the DHCPv6 PD Client, and the **ipv6 address autoconfig** command on the Stateless Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-12: PD Server / Subdelegation / Client / Stateless Client (multiple VLANs) configuration topology



Stateless_Client configuration

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client
!
interface vlan1
 ipv6 address autoconfig
```

PD_Server configuration

See the below configuration for a device with the hostname **PD_Server**:

```
hostname PD_Server
!
ipv6 local pool pd_vlan1 2001:db8:20::/48 56
!
ipv6 dhcp pool pool1
 prefix-delegation pool pd_vlan1
!
interface vlan1
 description to_PD_Subdelegation
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool1
!
ipv6 forwarding
```

PD_Subdelegation configuration

See the below configuration for a device with the hostname **PD_Subdelegation**:

```
hostname PD_Subdelegation
!
ipv6 local pool pdpool1 pool1 ::/56 64
!
ipv6 dhcp pool pool1
 prefix-delegation pool pdpool1

vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan1
 description to_PD_Server
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan2
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
interface vlan3
 description to_PD_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::11:0:0:0:0/64 eui64
 ipv6 dhcp server pool1
!
ipv6 forwarding
```

PD_Client configuration

See the below configuration for a device with the hostname **PD_Client**:

```
hostname PD_client
!
ipv6 dhcp pool pool1
!
vlan database
 vlan 3,4 state enable
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface vlan3
 description to_PD_Subdelegation
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan4
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
ipv6 forwarding
```

Configuring DHCPv6 Relay with Recursive PD Subdelegation

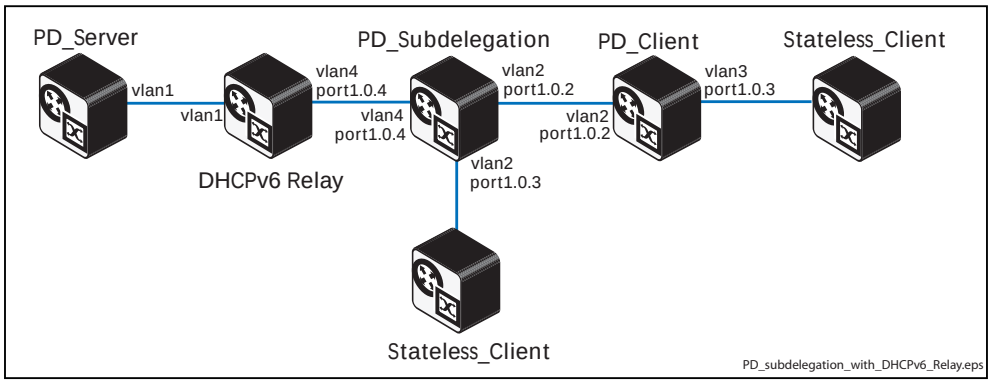
This example is similar to the previous example, **Configuring PD Subdelegation with SLAAC**, with the difference being that there is a DHCPv6 Relay device between the DHCPv6 PD Server device and the DHCPv6 PD subdelegation device.

For detailed command information, see the **prefix-delegation pool** and **ipv6 dhcp server** commands on the DHCPv6 PD Server and the DHCPv6 PD Subdelegation Server, the **ip dhcp-relay server-address** command on the DHCPv6 Relay, the **ipv6 dhcp client pd** command on the DHCPv6 PD Client, and the **ipv6 address autoconfig** command on the Stateless Client.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

Figure 91-13: DHCPv6 Server / Stateful Client configuration topology



Stateless_Client configuration

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client1
!
interface vlan1
 ipv6 address autoconfig
```

PD_Server configuration

See the below configuration for a device with the hostname **PD_Server**:

```
hostname PD_Server
!
ipv6 local pool pd_vlan2_relay 2001:db8:20::/48 56
!
ipv6 dhcp pool pool1
 prefix-delegation pool pd_vlan2_relay
 link-address 2001:db8:40::/64
!
interface vlan1
 description to_DHCPv6_Relay
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
ipv6 route 2001:db8:40::/64 fe80::200:cdff:fe29:a65f vlan1
```

DHCPv6_Relay configuration

See the below configuration for a device with the hostname **DHCPv6_Relay**:

```
hostname DHCPv6_Relay
!
vlan database
 vlan 4 state enable
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface vlan1
 description to_PD_Server
 ipv6 address 2001:db8:20::2/64
 ipv6 enable
!
interface vlan4
 description to_PD_Subdelegation
 ipv6 address 2001:db8:40::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ip dhcp-relay server-address 2001:db8:10::1 vlan1
!
ipv6 forwarding
```

PD_Subdelegation configuration

See the below configuration for a device with the hostname **PD_Subdelegation**:

```
hostname PD_Subdelegation
!
ipv6 local pool pdpool1 pool1 ::/56 64
!
ipv6 dhcp pool pool1
 prefix-delegation pool pdpool1
!
vlan database
 vlan 2,4 state enable
!
interface port1.0.2-port1.0.3
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface vlan4
 description to_DHCPv6_Relay
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan2
 description to_PD_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
```

PD_Client configuration

See the below configuration for a device with the hostname **PD_Client**:

```
hostname PD_Client
!
ipv6 dhcp pool pool1
!
vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan2
 description to_PD_Subdelegation
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan3
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
ipv6 forwarding
!
```

PD Subdelegation System Configuration

This section provides an example of a DHCPv6 PD sub-delegating system, involving multiple delegation pools, with device configuration output listed for each of the devices configured for PD sub-delegation:

- **Stateful_Client Configuration**
- **Stateless_Client Configuration**
- **PD_Subdelegation Configuration**
- **PD_Client Configuration**
- **PD_Server1 Configuration**
- **PD_Server2 Configuration**
- **DHCPv6_Relay Configuration**

Note the functionality for configured devices in this multiple delegation pool system:

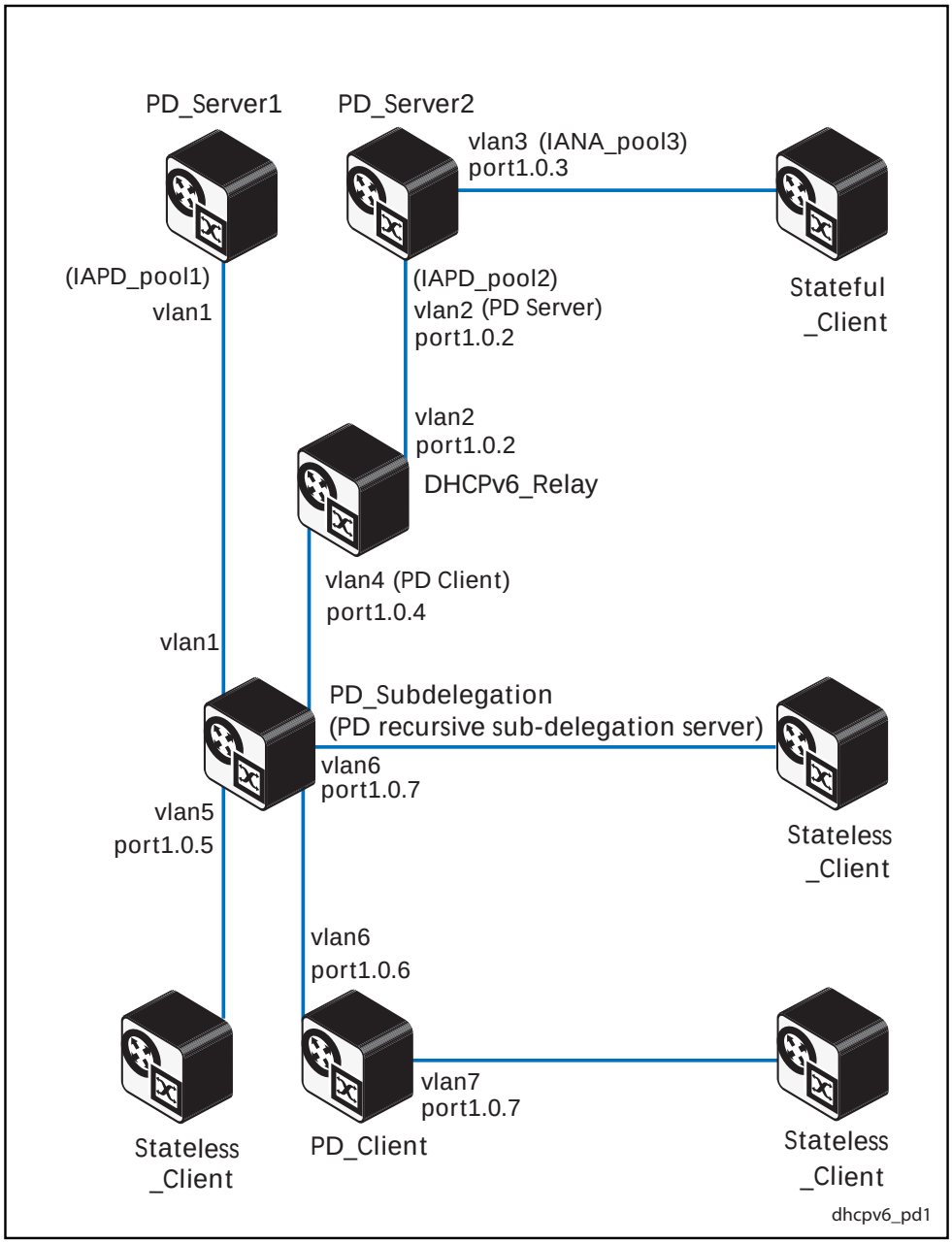
- **PD_Server1** is configured with a single IAPD delegation pool, **IAPD_pool1**.
- **PD_Server2** is configured with two pools; **IAPD_pool2** is used to delegate prefix information via intermediate **DHCPv6_Relay** to the subdelegation device via **vlan2**, and **IANA_pool3** is used to delegate address information to **Stateful_Client** via **vlan3**.
- **PD_Subdelegation** uses SLAAC to configure its upstream interfaces and acts as PD client on its upstream interfaces (**vlan1** and **vlan4**) that face each PD server.
- **PD_Subdelegation** downstream interface **vlan5** is configured based on the prefix delegated via the upstream interface **vlan1**.
- **PD_Subdelegation** downstream interface **vlan6** is configured based on the prefix delegated via upstream interface **vlan4**.
- Prefix information delegated via **PD_Client** interface **vlan4** is recursively sub-delegated via the downstream interface **vlan6** to **PD_Client**.
- Stateless clients auto configure themselves based on RAs.

Note **bold** configuration command entries show hostnames and interfaces.

Also note the default VLAN interface **vlan1** is applied to all ports on a device.

For the syntaxes, parameters, descriptions, defaults, and examples for all of the commands used in the following DHCPv6 configurations, refer to **Chapter 92, DHCP for IPv6 (DHCPv6) Commands** for the DHCPv6 Client Server PD commands and **Chapter 90, Dynamic Host Configuration Protocol (DHCP) Commands** for the DHCPv6 Relay commands.

Figure 91-14: PD sub-delegation system configuration example topology



Stateful_Client Configuration

See the below configuration for a device with the hostname **Stateful_Client**:

```
hostname Stateful_Client
!
interface vlan1
 ipv6 address dhcp
```

Stateless_Client Configuration

See the below configuration for a device with the hostname **Stateless_Client**:

```
hostname Stateless_Client
!
interface vlan1
 ipv6 address autoconfig
```

PD_Client Configuration

See the below configuration for a device with the hostname **PD_Client**:

```
hostname PD_Client
!
ipv6 dhcp pool pool1
!
vlan database
 vlan 6,7 state enable
!
interface port1.0.6
 switchport
 switchport mode access
 switchport access vlan 6
!
interface port1.0.7
 switchport
 switchport mode access
 switchport access vlan 7
!
interface vlan6
 description to_PD_Subdelegation
 ipv6 address autoconfig
 ipv6 dhcp client pd pool1
!
interface vlan7
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::/64 eui64
!
ipv6 forwarding
```

DHCPv6_Relay Configuration

See the following configuration for the device with the hostname **DHCPv6_Relay**:

```
hostname DHCPv6_Relay
!
vlan database
 vlan 2,4 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface vlan2
 description to_PD_Server2
 ipv6 address 2001:db8:20::2/64
 ipv6 enable
 no ipv6 nd suppress-ra
!
interface vlan4
 description to_PD_Subdelegation
 ipv6 address 2001:db8:40::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ip dhcp-relay server-address 2001:db8:20::1 vlan2
!
ipv6 forwarding
```

PD_Subdelegation Configuration

See the below configuration for a device with the hostname **PD_Subdelegation**:

```
hostname PD_Subdelegation
!
ipv6 local pool pdpool1 pool1 ::/56 64
ipv6 local pool pdpool2 pool2 ::/56 64
!
ipv6 dhcp pool pool1
 prefix-delegation pool pdpool1
!
ipv6 dhcp pool pool2
 prefix-delegation pool pdpool2
!
vlan database
 vlan 4,5,6 state enable
!
interface port1.0.4
 switchport
 switchport mode access
 switchport access vlan 4
!
interface port1.0.5
 switchport
 switchport mode access
 switchport access vlan 5
!
interface port1.0.6
 switchport
 switchport mode access
 switchport access vlan 6
!
interface vlan1
 description to_PD_Server1
 ipv6 address auto
 ipv6 dhcp client pd pool1
!
interface vlan4
 description to_DHCPv6_Relay
 ipv6 address auto
 ipv6 dhcp client pd pool2
!
interface vlan5
 description to_Stateless_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool1 ::1:0:0:0:0/64 eui64
 ipv6 nd prefix pool1 ::1:0:0:0:0/64
!
interface vlan6
 description to_PD_Client
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 address pool2 ::1/64
 ipv6 dhcp server pool2
!
ipv6 forwarding
```

PD_Server1 Configuration

See the following configuration for the device with the hostname **PD_Server1**:

```
hostname PD_Server1
!
ipv6 local pool pd_direct_vlan1 2001:db8:50::/48 56
!
ipv6 dhcp pool pool1
 prefix-delegation pool pd_direct_vlan1
!
interface vlan1
 description to_PD_Subdelegation
 ipv6 address 2001:db8:10::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool1
!
ipv6 forwarding
!
```

PD_Server2 Configuration

See the following configuration for the device with the hostname **PD_Server2**:

```
hostname PD_Server2
!
ipv6 local pool pd_relay_vlan2 2001:db8:60::/48 56
!
ipv6 dhcp pool pool2
 link-address 2001:db8:40::/64
 prefix-delegation pool pd_relay_vlan2
!
ipv6 dhcp pool pool3
 address prefix 2001:db8:30::/64
!
vlan database
 vlan 2,3 state enable
!
interface port1.0.2
 switchport
 switchport mode access
 switchport access vlan 2
!
interface port1.0.3
 switchport
 switchport mode access
 switchport access vlan 3
!
interface vlan2
 description to_DHCPv6_Relay
 ipv6 address 2001:db8:20::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool2
!
interface vlan3
 description to_Stateful_Client
 ipv6 address 2001:db8:30::1/64
 ipv6 enable
 no ipv6 nd suppress-ra
 ipv6 dhcp server pool3
!
ipv6 forwarding
!
ipv6 route 2001:db8:40::/64 fe80::200:cdff:fe29:a65f vlan2
!
```

Chapter 92: DHCP for IPv6 (DHCPv6) Commands

Command List 92.2
address prefix 92.3
address range 92.5
clear counter ipv6 dhcp-client 92.7
clear counter ipv6 dhcp-server 92.7
clear ipv6 dhcp binding 92.8
clear ipv6 dhcp client 92.9
dns-server (DHCPv6) 92.10
domain-name (DHCPv6) 92.11
ipv6 address (DHCPv6 PD) 92.12
ipv6 address dhcp 92.14
ipv6 dhcp client pd 92.15
ipv6 dhcp option 92.16
ipv6 dhcp pool 92.18
ipv6 dhcp server 92.19
ipv6 local pool 92.20
ipv6 nd prefix (DHCPv6) 92.22
link-address 92.24
option (DHCPv6) 92.26
prefix-delegation pool 92.29
show counter ipv6 dhcp-client 92.31
show counter ipv6 dhcp-server 92.32
show ipv6 dhcp 92.33
show ipv6 dhcp binding 92.34
show ipv6 dhcp interface 92.36
show ipv6 dhcp pool 92.38
sntp-address 92.40

Command List

This chapter provides an alphabetical reference for commands used to configure DHCPv6. For introductory information, see **Chapter 91, DHCP for IPv6 (DHCPv6) Introduction and Configuration**.

DHCPv6 is a network protocol used to configure IPv6 hosts with IPv6 addresses and IPv6 prefixes for an IPv6 network. DHCPv6 is used instead of SLAAC (Stateless Address Autoconfiguration) at sites where centralized management of IPv6 hosts is needed. IPv6 routers require automatic configuration of IPv6 addresses and IPv6 prefixes. DHCPv6 Prefix Delegation provides automatic configuration of IPv6 addresses and IPv6 prefixes.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

Note

The IPv6 addresses shown use the address space 2001:0db8::/32, defined in RFC 3849 for documentation purposes. These addresses should not be used for practical networks (other than for testing purposes) nor should they appear on any public network.

address prefix

Use this command in DHCPv6 Configuration mode to specify an address prefix for address assignment with DHCPv6 server pool configuration.

Use the **no** variant of this command to remove the address prefix from the DHCPv6 server pool.

Syntax

```
address prefix <ipv6-prefix/prefix-length> [lifetime {<valid-time>|
    infinite} {<preferred-time>|infinite}]

no address prefix <ipv6-prefix/prefix-length>
```

Parameter	Description
<ipv6-prefix/prefix-length>	Specify an IPv6 prefix and prefix length, The prefix length indicates the length of the IPv6 prefix assigned to the pool. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
`lifetime`	Optional. Specify a time period for the hosts to remember router advertisements (RAs). If you specify the optional `lifetime` parameter with this command then you must also specify a *valid-time* and a *preferred-time* value. See the **Usage** notes below this parameter table for a description of preferred and valid lifetimes and how these determine deprecated or invalid IPv6 addresses upon expiry.
<valid-time>	Specify a valid lifetime in seconds in the range <5-315360000>. The default valid lifetime is 2592000 seconds.
`infinite`	Specify an infinite valid lifetime or an infinite preferred lifetime, or both, when using this keyword.
<preferred-time>	Specify a preferred lifetime in seconds in the range <5-315360000>. The default preferred lifetime is 604800 seconds.

Mode DHCPv6 Configuration

Default The default valid lifetime is 2592000 seconds and the default preferred lifetime is 604800 seconds.

Usage This command creates a pool of prefixes from which addresses are assigned to clients on request, and allocates a network prefix from which the DHCPv6 Server leases addresses. This command is an alternative to using a range set using the **address range** command.

The DHCPv6 Server selects an IPv6 address from the range available allocated by the IPv6 prefix, randomly generating the suffix of the IPv6 address, with the specified preferred and valid lifetime leases. Leased IPv6 address are found in the DHCPv6 Server **REPLY** packet, which is located within the IANA (Identity Association for Non-temporary Addresses) IA address field in the **REPLY** message.

For more message information see the **DHCPv6 Messages** section and the **DHCPv6 Client and Server Identification** section in **Chapter 91, DHCP for IPv6 (DHCPv6) Introduction and Configuration**.

Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires.

Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.

An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.

Examples To add IPv6 address prefix `2001:0db8:1::/48` for DHCPv6 server pool configuration, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool pool1
awplus(config-dhcp6)# address prefix 2001:0db8:1::/48
```

To remove a configured IPv6 address prefix for DHCPv6 server pool configuration, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool pool1
awplus(config-dhcp6)# no address prefix 2001:0db8:1::/48
```

Related Commands
address range
ipv6 dhcp pool

Validation Commands
show ipv6 dhcp binding
show ipv6 dhcp pool

address range

Use this command in DHCPv6 Configuration mode to specify an address range for address assignment with DHCPv6 server pool configuration.

Use the **no** variant of this command to remove an address range from the DHCPv6 server pool.

Syntax

```
address range <first-ipv6-address> <last-ipv6-address>
   [lifetime {<valid-time>|infinite} {<preferred-time>|infinite}]

no address range <first-ipv6-address> <last-ipv6-address>
```

Parameter	Description
<first-ipv6-address>	Specify the first IPv6 address of the IPv6 address range, in hexadecimal notation in the format X:X::X:X.
<last-ipv6-address>	Specify the last IPv6 address of the IPv6 address range, in hexadecimal notation in the format X:X::X:X.
lifetime	Optional. Specify a time period for the hosts to remember router advertisements (RAs). If you specify this parameter then you must also specify a *valid-time* and a *preferred-time* value. See the **Usage** notes below this parameter table for a description of preferred and valid lifetimes and how these determine deprecated or invalid IPv6 addresses upon expiry.
<valid-time>	Specify a valid lifetime in seconds in the range <5-31536000>. The default valid lifetime is 2592000 seconds.
infinite	Specify an infinite valid lifetime or an infinite preferred lifetime, or both, when using this keyword.
<preferred-time>	Specify a preferred lifetime in seconds in the range <5-31536000>. The default preferred lifetime is 604800 seconds.

Default The default valid lifetime is 2592000 seconds and the default preferred lifetime is 604800 seconds.

Mode DHCPv6 Configuration

Usage Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires.

Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.

An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.

Examples

To add the IPv6 address range `2001:0db8:1::1` to `2001:0db8:1fff::1` for DHCPv6 server pool configuration, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool pool1
awplus(config-dhcp6)# address range 2001:0db8:1::1
2001:0db8:1fff::1
```

To remove a configured IPv6 address range for DHCPv6 server pool configuration, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool pool1
awplus(config-dhcp6)# no address range
```

Related Commands

address prefix
ipv6 dhcp pool

Validation Commands

show ipv6 dhcp binding
show ipv6 dhcp pool

clear counter ipv6 dhcp-client

Use this command in Privileged Exec mode to clear DHCPv6 client counters.

Syntax `clear counter ipv6 dhcp-client`

Mode Privileged Exec

Example To clear DHCPv6 client counters, use the following command:

```
awplus# clear counter ipv6 dhcp-client
```

Related Commands **show counter ipv6 dhcp-client**

clear counter ipv6 dhcp-server

Use this command in Privileged Exec mode to clear DHCPv6 server counters.

Syntax `clear counter ipv6 dhcp-server`

Mode Privileged Exec

Example To clear DHCPv6 server counters, use the following command:

```
awplus# clear counter ipv6 dhcp-server
```

Related Commands **show counter ipv6 dhcp-server**

clear ipv6 dhcp binding

Use this command in Privileged Exec mode to clear either a specific lease binding or the lease bindings as specified by the command parameters. The command will only take effect on dynamically allocated bindings, not statically configured bindings. This command clears binding entries on the DHCPv6 server binding table.

Syntax `clear ipv6 dhcp binding {ipv6 <prefix>|duid <DUID>|all|pool <name>}`

Mode Privileged Exec

Usage A specific binding may be deleted by **ipv6** address or **duid** address, or several bindings may be deleted at once using **all** or **pool**.

Note that if you specify to clear the **ipv6** or **duid** address of what is actually a static DHCPv6 binding, an error message is displayed. If **all** or **pool** are specified and one or more static DHCPv6 bindings exist within those addresses, any dynamic entries within those addresses are cleared but any static entries are not cleared.

The clear ipv6 dhcp binding command is used as a server function. A binding table entry on the DHCPv6 server is automatically:

- Created whenever a prefix is delegated to a client from the configuration pool.
- Updated when the client renews, rebinds, or confirms the prefix delegation.
- Deleted when the client releases all the prefixes in the binding, all prefix lifetimes have expired, or when a user runs the clear ipv6 dhcp binding command.

If the clear ipv6 dhcp binding command is used with the optional IPv6 address parameter, only the binding for the specified client is deleted. If the clear ipv6 dhcp binding command is used without the optional IPv6 address parameter, then all automatic client bindings are deleted from the DHCPv6 bindings table.

Example To clear all dynamic DHCPv6 server binding entries, use the command:

```
awplus# clear ipv6 dhcp binding all
```

Output **Figure 92-1: Example output from the clear ipv6 dhcp binding all command**

```
awplus#clear ipv6 dhcp binding all
% Deleted 1 entries
```

Related Commands **show ipv6 dhcp binding**

clear ipv6 dhcp client

Use this command in Privileged Exec mode to restart a DHCPv6 client on an interface.

Syntax `clear ipv6 dhcp client <interface>`

Parameter	Description
<interface>	Specify the interface name to restart a DHCPv6 client on.

Mode Privileged Exec

Example To restart a DHCPv6 client on interface vlan1, use the following command:

```
awplus# clear ipv6 dhcp client vlan1
```

Related Commands **show ipv6 dhcp binding**

dns-server (DHCPv6)

Use this command to add a Domain Name System (DNS) server to the DHCPv6 address pool you are configuring. You can use this command multiple times to create a list of DNS name servers available to the client. This sets the DNS server details using the pre-defined option 6. Note that if you add a user-defined option 6 using the **option (DHCPv6)** command on page 92.26, then you will override any settings created with this command.

Use the **no** variant of this command to remove either the specified DNS server or all DNS servers from the DHCPv6 pool.

Syntax

```
dns-server <ipv6-address>
no dns-server [<ipv6-address>]
```

Parameter	Description
<ipv6-address>	Specify an IPv6 address of the DNS server, in hexadecimal notation in the format `X:X::X:X`. This parameter is required when adding a DNS server to the DHCPv6 address pool. All DNS servers are removed from the DHCPv6 pool if you enter the `no dns-server` command without this parameter.

Mode DHCPv6 Configuration

Examples To add the DNS server with the assigned IPv6 address `2001:0db8:3000:3000::32` to the DHCPv6 server pool named `P2`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(dhcpv6-config)# dns-server 2001:0db8:3000:3000::32
```

To remove the DNS server with the assigned IPv6 address `2001:0db8:3000:3000::32` from the DHCPv6 server pool named `P2`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(dhcpv6-config)# no dns-server 2001:0db8:3000:3000::32
```

To remove all DNS servers from the DHCPv6 server pool named `P2`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(dhcpv6-config)# no dns-server
```

Related Commands

ipv6 dhcp pool
option (DHCPv6)
show ipv6 dhcp pool

domain-name (DHCPv6)

Use this command in DHCPv6 Configuration mode to add a domain name to the DHCPv6 server address pool you are configuring.

Use the **no** variant of this command to remove a domain name from the address pool.

Syntax

```
domain-name <domain-name>
no domain-name
```

Parameter	Description
<domain-name>	Specify the domain name you wish to assign the DHCPv6 server address pool. Valid characters are printable characters. If the name contains spaces then you must enclose it in "quotation marks".

Mode DHCPv6 Configuration

Usage This command specifies the domain name that a client should use when resolving host names using the Domain Name System, and sets the domain name details using the pre-defined option 15. Note that if you add a user-defined option 15 using the **option (DHCPv6)** command on page 92.26, then you will override any settings created with this command.

Examples To add the domain name `Engineering` to DHCPv6 server pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(dhcpv6-config)# domain-name Engineering
```

To remove the domain name `Engineering` from DHCPv6 server pool `P2`, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(dhcpv6-config)# no domain-name Engineering
```

Related Commands

dns-server (DHCPv6)
option (DHCPv6)
show ipv6 dhcp pool

ipv6 address (DHCPv6 PD)

Use this command in Interface Configuration mode for a VLAN interface to append an IPv6 address suffix to the IPv6 prefix provided by a DHCPv6 Prefix Delegation (PD) server.

Use the **no** variant of this command to remove the IPv6 address assigned and disable IPv6. Note that if no global addresses are left after removing the IPv6 address then IPv6 is disabled.

Syntax

```
ipv6 address [<ipv6-prefix-name>] <ipv6-addr/prefix-length> [eui64]
no ipv6 address [<ipv6-prefix-name>] <ipv6-addr/prefix-length>
    [eui64]
```

Parameter	Description
<ipv6-prefix-name>	The IPv6 prefix name advertised on the router advertisement message sent from the switch. The IPv6 prefix name is delegated from the DHCPv6 Server configured for DHCPv6 Prefix-Delegation.
<ipv6-addr/prefix-length>	Specifies the IPv6 address to be set, for example ::1/64. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
`[eui64]`	A method of automatically deriving the lower 64 bits of an IPv6 address, based on the switch's MAC address.

Mode Interface Configuration for a VLAN interface.

Usage When specifying the eui64 parameter, the interface identifier of the IPv6 address is derived from the MAC address of the switch.

See the **IPv6 EUI-64 Addressing** section in **Chapter 30, IPv6 Introduction** for further EUI-64 implementation information.

Examples To configure a PD prefix named `prefix1` on interface `vlan1` and then add an IPv6 address, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 dhcp client pd prefix1
awplus(config-if)# ipv6 address prefix1::1/64
```

In this example, the prefix will be assigned from the pool on the PD client. The host portion or suffix will be ::1 for the last 64 bits.

To configure a PD prefix named `prefix1` on interface `vlan1` and then add an IPv6 address using EUI-64 identifiers, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 dhcp client pd prefix1
awplus(config-if)# ipv6 address prefix1/64 eui64
```

In this example, the prefix will be assigned from the pool on the PD client. The host portion or suffix is created from the EUI-64 identifier of the interface for the last 64 bits.

To assign the IPv6 address `2001:0db8::a2/48` to the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 address 2001:0db8::a2/48
```

To remove the IPv6 address `2001:0db8::a2/48` from the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 address 2001:0db8::a2/48
```

To assign the **eui64** derived address in the prefix `2001:db8::/64` to VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 address 2001:0db8::/64 eui64
```

To remove the **eui64** derived address in the prefix `2001:db8::/32` from VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 address 2001:0db8::/64 eui64
```

Validation Commands

show running-config
show ipv6 dhcp binding
show ipv6 interface brief
show ipv6 route

Related Commands
ipv6 dhcp client pd
ipv6 dhcp pool
ipv6 local pool
ipv6 nd prefix (DHCPv6)
prefix-delegation pool

ipv6 address dhcp

Use this command in Interface Configuration mode to activate the DHCPv6 client on the interface that you are configuring. This allows the interface to use the DHCPv6 client to obtain its IPv6 configuration details from a DHCPv6 server on its connected network.

Use the **no** variant of this command to stop the interface from obtaining IPv6 configuration details from a DHCPv6 server.

The DHCPv6 client supports the following IP configuration options:

- Option 1 - the subnet mask for your switch.
- Option 3 - a list of default routers.
- Option 6 - a list of DNS servers. This list appends the DNS servers set on your switch with the **dns-server (DHCPv6)** command.
- Option 15 - a domain name used to resolve host names. This option replaces the domain name set with the **domain-name (DHCPv6)** command.
- Option 51 - lease expiration time.

Syntax
```
ipv6 address dhcp
no ipv6 address dhcp
```

Examples To set the interface `vlan10` to use DHCPv6 to obtain an IPv6 address, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config)# ipv6 enable
awplus(config-if)# ipv6 address dhcp
```

To stop the interface `vlan10` from using DHCPv6 to obtain its IPv6 address, use the commands:

```
awplus# configure terminal
awplus(config)# interface vlan10
awplus(config-if)# no ipv6 address dhcp
```

Related Commands
ipv6 address

Validation Commands
show running-config

ipv6 dhcp client pd

Use this command in Interface Configuration mode to enable the DHCPv6 client process and enable requests for prefix delegation through the interface that you are configuring.

Use the **no** variant of this command to disable requests for prefix delegation. This is the default setting.

For further information about DHCPv6 Prefix Delegation, which is used to automate the process of assigning prefixes, see **Chapter 91, DHCP for IPv6 (DHCPv6) Introduction and Configuration**.

Syntax

```
ipv6 dhcp client pd <prefix-name>

no ipv6 dhcp client pd
```

Parameter	Description
<prefix-name>	Specify an IPv6 general prefix name. Valid characters are any printable character. If the name contains spaces then you must enclose it in "quotation marks".

Mode Interface Configuration

Default Prefix delegation is disabled by default on an interface.

Usage Entering the **ipv6 dhcp client pd** command starts the DHCPv6 client process if not already running, and enables requests for prefix delegation through the interface on which the command is configured.

When prefix delegation is enabled and a prefix is acquired, the prefix is stored in the IPv6 prefix pool with an internal name defined by the required *<prefix-name>* placeholder parameter. The **ipv6 address** command can then refer to the prefixes stored in the IPv6 prefix pool.

Examples To enable prefix delegation with the prefix name `prefix-name` on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 dhcp client pd my-prefix-name
```

To disable prefix delegation on the VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 dhcp client pd
```

Related Commands
clear ipv6 dhcp client
ipv6 address (DHCPv6 PD)
ipv6 nd prefix (DHCPv6)
show ipv6 dhcp binding

ipv6 dhcp option

Use this command in Global Configuration mode to create a user-defined DHCPv6 option. You can then use this option when configuring a DHCPv6 server address pool, by using the **option (DHCPv6)** command.

Options with the same number as one of the pre-defined options override the standard option definition. The pre-defined options use the option numbers 1, 3, 6, 15, and 51.

Use the **no** variant of this command to remove either the specified user-defined option. This also removes user-defined options from the associated DHCPv6 server address pools.

Syntax

```
ipv6 dhcp option <1-254> [name <option-name>] [<option-type>]
no ipv6 dhcp option <1-254>|<option-name>
```

Parameter	Description	
<1-254>	The option number of the option. Options with the same number as one of the standard options overrides the standard option definition.	
<option-name>	Option name used to identify the option. You cannot use a number as the option name. Valid characters are any printable character. If the name contains spaces then you must enclose it in "quotation marks". Default: no default	
<option-type>	The option value. You must specify a value that is appropriate to the option type:	
	`asci`	An ASCII text string
	`hex`	A hexadecimal string. Valid characters are the numbers 0–9 and letters a–f. Embedded spaces are not valid. The string must be an even number of characters, from 2 and 256 characters long.
	`ipv6`	An IPv6 address or prefix that has hexadecimal notation in the format `HHHH:HHHH::HHHH:HHHH`. To create a list of IPv6 addresses, you must add each IPv6 address individually by using the option command multiple times.
	`integer`	A number from 0 to 4294967295.
	`flag`	A value that either sets (to 1) or unsets (to 0) a flag: **true**, **on**, or **enabled** will set the flag. **false**, **off** or **disabled** will unset the flag.

Mode Global Configuration

Examples To define a user-defined ASCII string option as option 66, without a name, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp option 66 ascii
```

To define a user-defined hexadecimal string option as option 46, with the name "tcpip-node-type", use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp option 46 name tcpip-node-type hex
```

To define a user-defined IP address option as option 175, with the name `special-address`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp option 175 name special-address ip
```

To remove the specific user-defined option with the option number `12`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 dhcp option 12
```

To remove the specific user-defined option with the option name `perform-router-discovery`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 dhcp option perform-router-discovery
```

Related Commands
dns-server (DHCPv6)
domain-name (DHCPv6)
option (DHCPv6)
show ipv6 dhcp

ipv6 dhcp pool

Use this command in Global Configuration mode to enter the DHCPv6 Configuration mode for the DHCPv6 server pool name as specified in the required command parameter. If the name specified is not associated with an existing pool, the switch will create a new pool with this name, then enter the configuration mode for the new pool.

Once you have entered the DHCPv6 configuration mode, all commands executed before the next **exit** command will apply to this pool.

You can create multiple DHCPv6 server pools on switches with multiple interfaces. This allows the switch to act as a DHCPv6 server on multiple interfaces to distribute different information to clients on the different networks.

Use the **no** variant of this command to delete the specific DHCPv6 pool.

Syntax

```
ipv6 dhcp pool <DHCPv6-poolname>
no ipv6 dhcp pool <DHCPv6-poolname>
```

Parameter	Description
<DHCPv6-poolname>	Description used to identify this DHCPv6 server pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”.

Mode Global Configuration

Usage All DHCPv6 prefix pool names must be unique. IPv6 prefix pools have a similar function to IPv4 address pools. Contrary to IPv4, a block of IPv6 addresses (an IPv6 address prefix) are assigned and not single IPv6 addresses. IPv6 prefix pools are not allowed to overlap.

Once a pool is configured, it cannot be changed. To change the configuration, you must remove then recreate a IPv6 prefix pool. All IPv6 prefixes already allocated are also freed.

Examples To create the DHCPv6 pool named `P2` and enter DHCPv6 configuration mode, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)#
```

To delete the DHCPv6 pool named `P2`, use the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 dhcp pool P2
```

Related Commands
ipv6 local pool
option (DHCPv6)
prefix-delegation pool
show ipv6 dhcp binding
show ipv6 dhcp pool

ipv6 dhcp server

Use this command in Interface Configuration mode to enable DHCPv6 server for the current IPv6 configured interface to use the specified DHCPv6 server pool name.

The DHCPv6 server service listens for DHCPv6 requests on the IPv6 configured interface. The DHCPv6 server service does not run on interfaces without IPv6 configured on them.

Use the **no** variant of this command to disable the DHCPv6 server.

Syntax

```
ipv6 dhcp-server [<DHCPv6-poolname>]

no ipv6 dhcp-server
```

Parameter	Description
<DHCPv6-poolname>	Specify a named DHCPv6 server pool as defined with the **ipv6 dhcp pool** command. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”.

Mode Interface Configuration

Usage The **ipv6 dhcp server** command enables the DHCPv6 service on a specified interface using the pool for prefix delegation and configuration through the specified interface.

Note that DHCPv6 client, DHCPv6 server and DHCPv6 relay are mutually exclusive on an interface. When one of the DHCPv6 functions is enabled on an interface then another DHCPv6 function cannot be enabled on the same interface.

Examples To enable the DHCPv6 server service and use the DHCPv6 pool named `P2` on VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 dhcp server P2
```

To disable the DHCPv6 server on VLAN interface `vlan2`, use the following commands:

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# no ipv6 dhcp server
```

Related Commands

ipv6 dhcp pool
show ipv6 dhcp binding
show ipv6 dhcp pool

ipv6 local pool

Use this command in Global Configuration mode to configure a local DHCPv6 server prefix delegation pool specifying a poolname and a prefix/prefix length. You can optionally exclude the locally assigned prefix from the pool with the **exclude-local-prefix** keyword.

Use the **no** variant of this command to remove a local DHCPv6 server prefix delegation pool specifying the poolname.

Syntax

```
ipv6 local pool <DHCPv6-poolname> <delegated-prefix-name>
   <ipv6-prefix/prefix-length> <assigned-length>
   [exclude-local-prefix]

no ipv6 local pool
```

Parameter	Description
<DHCPv6-poolname>	Description used to identify this DHCPv6 server pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”.
<delegated-prefix-name>	Description used to identify the delegated prefix name from the parent PD (Prefix Delegation) server. If the name contains spaces then you must enclose it in “quotation marks”.
<ipv6-prefix/prefix-length>	Specify an IPv6 prefix and prefix length. The prefix length indicates the length of the IPv6 prefix assigned to the pool. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.
<assigned-length>	Specify an IPv6 prefix length assigned to the user from the pool in the range <1-128>. Note that the value of the *assigned-length* parameter entered cannot be less than or equal to the *prefix-length* parameter value entered. An assigned length must be longer than a prefix length.
exclude-local-prefix	Optional. Specify this keyword to exclude the locally assigned prefix from the pool.

Default No DHCPv6 server prefix delegation pool is configured by default.

Mode Global Configuration

Usage All IPv6 prefix pool names must be unique. IPv6 prefix pools have a similar function to IPv4 address pools. Contrary to IPv4, a block of IPv6 addresses (an IPv6 address prefix) are assigned and not single IPv6 addresses. IPv6 prefix pools are not allowed to overlap.

Once a pool is configured, it cannot be changed. To change the configuration, you must remove then recreate a IPv6 prefix pool. All IPv6 prefixes already allocated are also freed.

Examples To create a local DHCPv6 local pool named `P2` with the IPv6 prefix and prefix length `2001:0db8::/32` with an assigned length of `64`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 local pool P2 2001:0db8::/32 64
```

To remove a configured DHCPv6 local pool, use the following commands:

```
awplus# configure terminal
awplus(config)# no ipv6 local pool
```

Related Commands **ipv6 dhcp pool**
show ipv6 dhcp pool

ipv6 nd prefix (DHCPv6)

Use this command in Interface Configuration mode for a VLAN interface to specify IPv6 RA (Router Advertisement) prefix information generated from the DHCPv6 Server for DHCPv6 Prefix-Delegation for the VLAN interface.

Use the **no** variant of this command to remove IPv6 RA prefix information from the DHCPv6 Server for DHCPv6 Prefix-Delegation for the interface. Use the **all** parameter with the **no** variant of this command to remove all prefix names and all prefixes for an interface.

Syntax

```
ipv6 nd prefix <ipv6-prefix-name> <ipv6-prefix/length>
   {<valid-lifetime>|infinite} {<preferred-lifetime>|infinite}
   {off-link|no-autoconfig}

no ipv6 nd prefix {<ipv6-prefix-name>|<ipv6-prefix/length>|all}
```

Parameter	Description
<ipv6-prefix-name>	The IPv6 prefix name advertised on the router advertisement message sent from the switch. The IPv6 prefix name is delegated from the DHCPv6 Server configured for DHCPv6 Prefix-Delegation.
<ipv6-prefix/length>	The IPv6 prefix and prefix length advertised on the router advertisement message sent from the switch. The IPv6 address prefix uses the format X:X::/prefix-length. The prefix-length is usually set between 0 and 64..
<valid-lifetime>	The the period during which the specified IPv6 address prefix is valid. This can be set to a value between 5 and 315360000 seconds. Note that this period should be set to a value greater than that set for the prefix preferred-lifetime. See the **Usage** notes after this parameter table for a description of valid lifetime and how it determines invalid IPv6 addresses upon expiry.
`infinite`	Specifying this keyword instead of entering a value for the *<valid-lifetime>* parameter applies an infinite valid lifetime.
<preferred-lifetime>	Specifies theIPv6 prefix preferred lifetime. This is the period during which the IPv6 address prefix is considered current. Set this to a value between 0 and 315360000 seconds. Note that this period should be set to a value less than that set for the prefix valid-lifetime. See the **Usage** notes after this parameter table for a description of preferred lifetime and how it determines deprecated IPv6 addresses upon expiry.
`infinite`	Specifying this keyword instead of entering a value for the *<preferred-lifetime>* parameter applies an infinite valid lifetime.
`off-link`	Specify the IPv6 prefix off-link flag.
`no-autoconfig`	Specify the IPv6 prefix no autoconfiguration flag. Setting this flag indicates that the prefix is not to be used for autoconfiguration.
`all`	Specify all prefix names and all prefixes are removed when used with the **no** variant of this command.

Mode Interface Configuration for a VLAN interface.

Usage This command specifies the IPv6 prefix flags that are advertised by the router advertisement message.

Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires.

Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.

An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.

Examples The following example configures the switch to issue RAs (Router Advertisements) on the VLAN interface `vlan4`, and advertises the DHCPv6 prefix name `prefix1` and the IPv6 address prefix of `2001:0db8::/32`.

```
awplus# configure terminal
awplus(config)# interface vlan2
awplus(config-if)# ipv6 enable
awplus(config-if)# ipv6 dhcp client pd prefix1
awplus(config-if)# ipv6 nd prefix prefix1 2001:0db8::/32
```

The following example resets router advertisements on the VLAN interface `vlan4`, so the address prefix of `2001:0db8::/32` is not advertised from the switch.

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd prefix 2001:0db8::/32
```

The following example removes all prefix names and prefixes from VLAN interface `vlan4`:

```
awplus# configure terminal
awplus(config)# interface vlan4
awplus(config-if)# no ipv6 nd prefix all
```

Related Commands
ipv6 address (DHCPv6 PD)
ipv6 dhcp client pd
ipv6 dhcp pool
ipv6 local pool
prefix-delegation pool
show ipv6 dhcp binding

link-address

Use this command in DHCPv6 Configuration mode to specify a link-address prefix within a DHCPv6 Server pool.

Note that you can only configure one link address per DHCPv6 pool. Configuring another link address in the same DHCPv6 pool overwrites the previously configured link address.

Use the **no** variant of this command to remove the link-address prefix from the DHCPv6 Server pool.

Syntax

```
link-address <ipv6-prefix/prefix-length>

no link-address
```

Parameter	Description
<ipv6-prefix/prefix-length>	Specify an IPv6 prefix and prefix length. The prefix length indicates the length of the IPv6 prefix assigned to the pool. The IPv6 address uses the format X:X::X:X/Prefix-Length. The prefix-length is usually set between 0 and 64.

Default No DHCPv6 Server pool configuration link address prefix is configured by default.

Mode DHCPv6 Configuration

Usage Link addresses are configured in DHCPv6 Server address pools when there are remote clients that communicate via intermediate relay(s).

RELAY-FORW and **RELAY-REPL** relay packets contain the requesting link address source.

This command is used to match incoming requests from PD (Prefix Delegation) clients (received via an intermediate relay) to a configured delegation pool.

When an address on the incoming interface of the DHCPv6 server or a link address set in the incoming delegation request packet from the prefix delegation client matches the link-address prefix configured in the delegation pool, the DHCPv6 server is able to match and use the appropriate delegation pool for relayed delegation request messages.

If there is no match between incoming delegation request packets from the prefix delegation client and the link-address prefix configured in the delegation pool, the DHCPv6 Server does not delegate an IPv6 prefix to the requesting device.

The link address should be set to the network prefix where the prefix delegation client resides. The prefix delegation server will also need a forwarding path (IPv6 route) back to the network prefix where the prefix delegation client resides.

See the application of this command in the **Configuring DHCPv6 Prefix Delegation** section of **Chapter 91, DHCP for IPv6 (DHCPv6) Introduction and Configuration** and refer to the topology shown.

Examples To configure the IPv6 prefix and prefix length `2001:0db8:1::/48` as the link address for pool `P2`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# address prefix 2001:0db8:2::/48
awplus(config-dhcp6)# link-address 2001:0db8:1::/48
```

To remove the link address, use the commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# no link-address
```

Related Commands **ipv6 dhcp pool**
show ipv6 dhcp pool

option (DHCPv6)

Use this command in DHCPv6 Configuration mode to add a user-defined option to the DHCPv6 prefix pool you are configuring. For the **hex**, **integer**, and **flag** option types, if the option already exists, the new option overwrites the existing option's value.

Use the **no** variant of this command to remove the specified user-defined option from the DHCPv6 server pool, or to remove all user-defined options from the DHCPv6 server pool.

Syntax

```
option [<1-254>|<option-name>] <option-value>
no option [<1-254>|<option-value>]
```

Parameter	Description	
<1-254>	The option number of the option. Options with the same number as one of the standard options overrides the standard option definition.	
<option-name>	Option name associated with the option.	
<option-value>	The option value. You must specify a value that is appropriate to the option type:	
	`hex`	A hexadecimal string. Valid characters are the numbers 0–9 and letters a–f. Embedded spaces are not valid. The string must be an even number of characters, from 2 and 256 characters long.
	`ipv6`	An IPv6 prefix that has the hexadecimal `X:X::X:X` notation. To create a list of IPv6 prefixes, you must add each IPv6 prefix individually using this command multiple times.
	`integer`	A number from 0 to 4294967295.
	`flag`	A value of either true, on, or enabled to set the flag, or false, off or disabled to unset the flag.

Mode DHCPv6 Configuration

Usage You must define a DHCPv6 option using the **ipv6 dhcp option** command before using the **option (DHCPv6)** command.

Note that options with an **ipv6** type can hold a list of IPv6 prefix (i.e. entries that have the `X:X::X:X` address format), so if the option already exists in the pool, then the new IP address is added to the list of existing IPv6 prefixes. Also note options with the same number as one of the pre-defined options override the standard option definition. The pre-defined options use the option numbers 1, 3, 6, 15, and 51.

Examples To add the IPv6 type option named `sntp-server-addr` to the pool `P2` and give the option the value `ipv6`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp option 22 name
sntp_server_addr ipv6
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# option sntp_server_addr ipv6
```

To add the ASCII-type option named `tftp-server-name` to the pool `P2` and give the option the value `server1`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# option tftp-server-name server1
```

To add the hex-type option named `tcpip-node-type` to the pool `P2` and give the option the value `08af`, use the following commands:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# option tcpip-node-type 08af
```

To add multiple IP addresses for the ip-type option `175`, use the following commands:

```
awplus(config-dhcp6)# option 175 2001:0db8:3001::/64
awplus(config-dhcp6)# option 175 2001:0db8:3002::/64
awplus(config-dhcp6)# option 175 2001:0db8:3003::/64
```

To add the option `179` to a pool, and give the option the value `123456`, use the following command:

```
awplus(config-dhcp6)# option 179 123456
```

To add a user-defined flag option with the name `perform-router-discovery`, use the following command:

```
awplus(config-dhcp6)# option perform-router-discovery yes
```

To clear all user-defined options from a DHCP address pool, use the following command:

```
awplus(config-dhcp6)# no option
```

To clear a user-defined option, named `tftp-server-name`, use the following command:

```
awplus(config-dhcp6)# no option tftp-server-name
```

Related Commands

dns-server (DHCPv6)
ipv6 dhcp option
ipv6 dhcp pool
show ipv6 dhcp pool

prefix-delegation pool

Use this command in DHCPv6 Configuration mode to add a DHCPv6 server prefix-delegation pool entry to the current DHCPv6 pool configuration. You must define a DHCPv6 server prefix-delegation pool using the **ipv6 dhcp pool** command before using this command.

Use the **no** variant of this command to remove a DHCPv6 server prefix-delegation pool from the current DHCPv6 pool configuration.

Syntax

```
prefix-delegation pool <DHCPv6-poolname> [lifetime {<valid-time>|
    infinite} {<preferred-time>|infinite}]

no prefix-delegation pool <DHCPv6-poolname>
```

Parameter	Description
<DHCPv6-poolname>	Description used to identify this DHCPv6 server pool. Valid characters are any printable character. If the name contains spaces then you must enclose it in “quotation marks”.
`lifetime`	Optional. Specify a time period for the hosts to remember router advertisements (RAs). If you specify this parameter then you must also specify a *valid-time* and a *preferred-time* value. See the **Usage** notes below this parameter table for a description of preferred and valid lifetimes and how these determine deprecated or invalid IPv6 addresses upon expiry.
<valid-time>	Specify a valid lifetime in seconds in the range <5-315360000>.
`infinite`	Specify an infinite valid lifetime or an infinite preferred lifetime, or both, when using this keyword.
<preferred-time>	Specify a valid lifetime in seconds in the range <5-315360000>.

Default No IPv6 local prefix pool is specified by default.

Mode DHCPv6 Configuration

Usage The DHCPv6 server assigns prefixes dynamically from an IPv6 local prefix pool, which is configured using the **ipv6 local pool** command and is associated with a DHCPv6 configuration pool using this command. When the server receives a prefix request from a client, it attempts to obtain unassigned prefixes from the pool. After the client releases the previously assigned prefixes, the server returns the prefixes to the pool for reassignment.

Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires.

Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.

An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.

Example This example adds DHCPv6 Prefix Delegation pool `pd_pool1` to DHCPv6 pool `pool1`:

```
awplus# configure terminal
awplus(config)# ipv6 local pool pd_pool1 2001:0db8::/48
56
awplus(config)# ipv6 dhcp pool pool1
awplus(config-dhcp6)# prefix-delegation pool pd_pool1
```

Related Commands
ipv6 dhcp pool
ipv6 local pool
show ipv6 dhcp pool

show counter ipv6 dhcp-client

Use this command in User Exec or Privilege Exec mode to show DHCPv6 client counter information. See **show counter ipv6 dhcp-server** for DHCPv6 server information.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show counter ipv6 dhcp-client`

Mode User Exec and Privileged Exec

Example To display the DHCPv6 client counter information, use the command:

```
awplus# show counter ipv6 dhcp-client
```

Output **Figure 92-2: Example output from the show counter ipv6 dhcp-client command**

```
awplus#show counter ipv6 dhcp-client
SOLICIT out              ......... 20
ADVERTISE in             ......... 12
REQUEST out              ......... 1
CONFIRM out              ......... 0
RENEW out                ......... 0
REBIND out               ......... 0
REPLY in                 ......... 0
RELEASE out              ......... 0
DECLINE out              ......... 0
INFORMATION-REQUEST out  ......... 0
```

Table 92-1: Parameters in the output of the show counter ipv6 dhcp-client command

Parameter	Description
`SOLICIT out`	Displays the count of SOLICIT messages sent by the DHCPv6 client.
`ADVERTISE in`	Displays the count of ADVERTISE messages received by the DHCPv6 client.
`REQUEST out`	Displays the count of REQUEST messages sent by the DHCPv6 client.
`CONFIRM out`	Displays the count of CONFIRM messages sent by the DHCPv6 client.
`RENEW out`	Displays the count of RENEW messages sent by the DHCPv6 client.
`REBIND out`	Displays the count of REBIND messages sent by the DHCPv6 client.
`REPLY in`	Displays the count of REPLY messages received by the DHCPv6 client.
`RELEASE out`	Displays the count of RELEASE messages sent by the DHCPv6 client.
`DECLINE out`	Displays the count of DECLINE messages sent by the DHCPv6 client.
`INFORMATION-REQUEST out`	Displays the count of INFORMATION-REQUEST messages sent by the DHCPv6 client.

Related Commands **show counter ipv6 dhcp-server**

show counter ipv6 dhcp-server

Use this command in User Exec or Privileged Exec mode to show DHCPv6 server counter information. See **show counter ipv6 dhcp-client** for DHCPv6 client information.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show counter ipv6 dhcp-server`

Mode User Exec and Privileged Exec

Example To display the DHCPv6 server counter information, use the command:

```
awplus# show counter ipv6 dhcp-server
```

Output **Figure 92-3: Example output from the show counter ipv6 dhcp-server command**

```
awplus#show counter ipv6 dhcp-server
SOLICIT in               ......... 20
ADVERTISE out            ......... 12
REQUEST in               ......... 1
CONFIRM in               ......... 0
RENEW in                 ......... 0
REBIND in                ......... 0
REPLY out                ......... 0
RELEASE in               ......... 0
DECLINE in               ......... 0
INFORMATION-REQUEST in   ......... 0
```

Table 92-2: Parameters in the output of the show counter ipv6 dhcp-server command

Parameter	Description
`SOLICIT in`	Displays the count of SOLICIT messages received by the DHCPv6 server.
`ADVERTISE out`	Displays the count of ADVERTISE messages sent by the DHCPv6 server.
`REQUEST in`	Displays the count of REQUEST messages received by the DHCPv6 server.
`CONFIRM in`	Displays the count of CONFIRM messages received by the DHCPv6 server.
`RENEW in`	Displays the count of RENEW messages received by the DHCPv6 server.
`REBIND in`	Displays the count of REBIND messages received by the DHCPv6 server.
`REPLY out`	Displays the count of REPLY messages sent by the DHCPv6 server.
`RELEASE in`	Displays the count of RELEASE messages received by the DHCPv6 server.
`DECLINE in`	Displays the count of DECLINE messages received by the DHCPv6 server.
`INFORMATION-REQUEST in`	Displays the count of INFORMATION-REQUEST messages received by the DHCPv6 server

Related Commands **show counter ipv6 dhcp-client**

show ipv6 dhcp

Use this command in User Exec or Privileged Exec mode to show the DHCPv6 unique identifier (DUID) configured on your switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 dhcp`

Mode User Exec and Privileged Exec

Usage The DUID is based on the link-layer address for both DHCPv6 client and DHCPv6 server identifiers. The switch uses the MAC address from the lowest interface number for the DUID.

The DUID is used by a DHCPv6 client to obtain an IPv6 address from a DHCPv6 server. A DHCPv6 server compares the DUID with its database of DUIDs and sends configuration data for an IPv6 address plus the preferred and valid lease time values to a DHCPv6 client.

Example To display the DUID configured on your switch, use the command:

```
awplus# show ipv6 dhcp
```

Output **Figure 92-4: Example output from the show ipv6 dhcp command**

```
awplus#show ipv6 dhcp
DHCPv6 Server DUID: 0001000117ab6876001577f7ba23
```

Related Commands **ipv6 address dhcp**

show ipv6 dhcp binding

Use this command in User Exec or Privileged Exec mode to show the IPv6 address entries that the DHCPv6 server leases to DHCPv6 clients. Note that applying this command with the optional `summary` keyword parameter displays the number of addresses per pool, but not the address or prefix entries per pool.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 dhcp binding [summary]`

Parameter	Description
`summary`	Optional. Specify the `summary` keyword to display summarized information for DHCPv6 server leases to client nodes, displaying the number of address entries per pool, not the addresses or prefixes.

Mode User Exec and Privileged Exec

Example 1 To display the total DHCPv6 leasing address entries for all pools, use the command:

```
awplus# show ipv6 dhcp binding summary
```

Output **Figure 92-5: Example output from the show ipv6 dhcp binding summary command**

```
awplus# show ipv6 dhcp binding summary
Pool Name                  Number of Leased Addresses
-----------------------------------------------------
ia-na1                     3
ia-pd1                     5

Total in all Pools:        8
```

Table 92-3: Parameters in the output of the show ipv6 dhcp binding summary command

Parameter	Description
`Pool Name`	Displays a list of all the pool names.
`Number of Leased Addresses`	Displays the number of leased address entries for the pool.
`Total in all Pools`	Displays the total number of leased address entries for all pools.

Example 2 To display addresses, prefixes, and lifetimes for all DHCPv6 leasing entries by pool, enter:

```
awplus# show ipv6 dhcp binding
```

Output **Figure 92-6: Example output from the show ipv6 dhcp binding command**

```
awplus#show ipv6 dhcp binding
Pool ia-na1
 Address 2002:0:3c0::1
         client IAID 77f7ba23, DUID 0001000117c4bbb4001577f7ba23
         preferred lifetime 604800, valid lifetime 2592000
         starts  at 20 Aug 2012 18:38:29
         expires at 19 Sep 2012 18:38:29
Pool ia-pd1
 Prefix  2002:0:3c0::/42
         client IAID 77f7ba23, DUID 0001000117c4bbb4001577f7ba23
         preferred lifetime 604800, valid lifetime 2592000
         starts  at 20 Aug 2012 18:38:29
         expires at 19 Sep 2012 18:38:29
```

Table 92-4: Parameters in the output of the show ipv6 dhcp binding command

Parameter	Description
`Address`	Address delegated to the indicated IAID and DUID. See the IAID and DUID descriptions below for further information.
`Prefix`	Prefix delegated to the indicated IAID and DUID. See the IAID and DUID descriptions below for further information.
`DUID`	DHCPv6 unique identifier (DUID) (see RFC 3315). Each DHCPv6 client has as DUID. DHCPv6 servers use DUIDs to identify clients for the association of IAs (Identity Associations) with DHCPv6 clients. DHCPv6 clients use DUIDs to identify a DHCPv6 server.
`IAID`	Identify Association Identifier (IAID) (see RFC 3315). IAIDs are identifiers for IAs (Identity Associations), where an IA is a collection of IPv6 addresses assigned to a DHCPv6 client. Each IA has an associated IAD. Each DHCPv6 client may have more than one IA assigned to it. Each IA holds one type of address.
`preferred lifetime`	The preferred lifetime setting in seconds for the specified IAID and DUID. Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires. Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected.
`valid lifetime`	The valid lifetime setting in seconds for the specified IAID and DUID. An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.
`starts at`	The date and time at which the valid lifetime expires.
`expires at`	The date and time at which the valid lifetime expires.

Related Commands
clear ipv6 dhcp binding
ipv6 dhcp pool
show ipv6 dhcp pool

show ipv6 dhcp interface

Use this command in User Exec or Privileged Exec mode to display DHCPv6 information for a specified interface, or all interfaces when entered without the interface parameter.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 dhcp interface [<interface-name>]`

Parameter	Description
<interface-name>	Optional. Specify the name of the interface to show DHCPv6 information about. Omit this optional parameter to display DHCPv6 information for all interfaces DHCPv6 is configured on.

Mode User Exec and Privileged Exec

Example 1 To display DHCPv6 information for all interfaces DHCPv6 is configured on, use the command:

```
awplus# show ipv6 dhcp interface
```

Output **Figure 92-7: Example output from the show ipv6 dhcp interface command**

```
awplus# show ipv6 dhcp interface
vlan1 is in client mode
 Address 1001::3c0:1
         preferred lifetime 9000, valid lifetime 5000
         starts  at 20 Jan 2012 09:21:35
         expires at 20 Jan 2012 10:25:32
vlan2 is in client (Prefix-Delegation) mode
 Prefix name pd1
         prefix 2002:0:3c0::/42
         preferred lifetime 604800, valid lifetime 2592000
         starts  at 20 Aug 2012 09:21:33
         expires at 19 Sep 2012 09:21:33
vlan3 is in server mode
 Using pool  : pool-1;
 Preference  : 0
```

Example 2 To display DHCPv6 information for interface `vlan2`, use the command:

```
awplus# show ipv6 dhcp interface vlan2
```

Output **Figure 92-8: Example output from the show ipv6 dhcp interface vlan2 command**

```
awplus# show ipv6 dhcp interface vlan2
vlan2 is in client (Prefix-Delegation) mode
 Prefix name pd1
         prefix 2002:0:3c0::/42
         preferred lifetime 604800, valid lifetime 2592000
         starts  at 20 Aug 2012 09:21:33
         expires at 19 Sep 2012 09:21:33
```

Table 92-5: Parameters in the output of the show counter dhcp-client command

Parameter	Description
`is in server/ client/(Prefix-Delegation) mode`	Displays whether the specified interface is in server or client mode and whether prefix-delegation is applied to an interface.
`Address`	Displays the address of the DHCPv6 server on the interface.
`Prefix name`	Displays the IPv6 general prefix pool name, where prefixes are stored for the interface.
`Using pool`	Displays the name of the pool used by the interface.
`Preference`	Displays the preference value for the DHCPv6 server.

Related Commands **ipv6 dhcp client pd**

show ipv6 dhcp pool

Use this command in User Exec or Privileged Exec mode to display the configuration details and system usage of the DHCPv6 address pools configured on the switch.

For information on output options, see **"Controlling "show" Command Output" on page 1.37**.

Syntax `show ipv6 dhcp pool [<DHCPv6-address-pool-name>]`

Parameter	Description
<DHCPv6-address-pool-name>	Name of a specific DHCPv6 address pool. This displays the configuration of the specified DHCPv6 address pool only.

Mode User Exec and Privileged Exec

Example

```
awplus# show ipv6 dhcp pool
```

Output **Figure 92-9: Example output from the show ipv6 dhcp pool command**

```
awplus# show ipv6 dhcp pool
DHCPv6 Pool: ia-na
 Address Prefix  : 1001::/64
         Lifetime: 2592000(valid), 604800(preferred)
 DNS Server: 2001::1
 DNS Server: 2001::2
 Domain Name: example.com
 Domain Name: example.co.jp
 SNTP Server: 2001::5
 SNTP Server: 2001::6
 Option Code : 150
        Value: [ASCII] test-test
DHCPv6 Pool: ia-pd
 PD Pool Name: pd1
    Prefix   : 2002::/38-42
    Lifetime : 2592000(valid), 604800(preferred)
```

Table 92-6: Parameters in the output of the show ipv6 dhcp pool command

Parameter	Description
`DHCPv6 Pool`	Name of the DHCPv6 pool.
`Address Prefix`	Address prefix to the DHCPv6 pool.

Table 92-6: Parameters in the output of the show ipv6 dhcp pool command(cont.)

Parameter	Description
`Address Lifetime`	Valid and preferred lifetimes to the DHCPv6 pool. Preferred IPv6 addresses or prefixes are available to interfaces for unrestricted use and are deprecated when the preferred timer expires. Deprecated IPv6 addresses and prefixes are available for use and are discouraged but not forbidden. A deprecated address or prefix should not be used as a source address or prefix, but packets sent from deprecated addresses or prefixes are delivered as expected. An IPv6 address or prefix becomes invalid and is not available to an interface when the valid lifetime timer expires. Invalid addresses or prefixes should not appear as the source or destination for a packet.
`DNS Server`	IPv6 address of the DNS Server
`Domain name`	URL for the domain name.
`SNTP Server`	IPv6 address of the SNTP (Simple Network Time Protocol) Server.
`Option Code`	DHCP Option code (see RFC 2132).
`Option Value`	DHCP Option value type (see RFC 2132).

Related Commands **ipv6 dhcp pool**

sntp-address

Use this command in DHCPv6 Configuration mode to add an SNTP Server IPv6 address to a DHCPv6 Server pool.

Use the **no** variant of this command to remove an SNTP Server IPv6 address from a DHCPv6 Server pool.

Syntax

```
sntp-address <ipv6-address>
no sntp-address <ipv6-address>
```

Parameter	Description
<ipv6-address>	Specify an SNTP Server IPv6 address, in hexadecimal notation in the format `X:X::X:X`.

Mode DHCPv6 Configuration

Examples The following example adds an SNTP Server IPv6 address of `2001:0db8::/32` to the DHCPv6 pool named `P2`:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# sntp-address 2001:0db8::/32
```

The following example removes an SNTP Server IPv6 address of `2001:0db8::/32` to the DHCPv6 pool named `P2`:

```
awplus# configure terminal
awplus(config)# ipv6 dhcp pool P2
awplus(config-dhcp6)# no sntp-address 2001:0db8::/32
```

Related Commands

dns-server (DHCPv6)
domain-name (DHCPv6)
option (DHCPv6)
show ipv6 dhcp pool

Chapter 93: SNMP Introduction

Introduction 93.2
Network Management Framework 93.2
Structure of Management Information 93.4
Names 93.5
Instances 93.6
Syntax 93.7
Access 93.7
Status 93.7
Description 93.7
The SNMP Protocol 93.8
SNMP Versions 93.8
SNMP Messages 93.9
Polling versus Event Notification 93.9
Message Format for SNMPv1 and SNMPv2c 93.10
SNMP Communities (Version v1 and v2c) 93.11
SNMPv3 Entities 93.11
SNMPv3 Message Protocol Format 93.12
SNMPv1 and SNMPv2c 93.13
SNMP MIB Views for SNMPv1 and SNMPv2c 93.13
SNMP Communities 93.13
Configuration Example (SNMPv1 and v2) 93.15
SNMPv3 ..... 93.18
SNMP MIB Views for SNMPv3 ..... 93.18
SNMP Groups 93.18
SNMP Users 93.18
Configuration Example (SNMPv3) 93.19
Using SNMP to Manage Files and Software 93.20
Copy a File to or from a TFTP Server 93.20
Upgrade Software and Configuration Files 93.22

Introduction

The Simple Network Management Protocol (SNMP) is the network management protocol of choice for the Internet and IP-based internetworks.

This chapter describes the main features of SNMP Version 1 (SNMPv1), SNMP Version 2c (SNMPv2c) and Version 3 (SNMPv3). It also describes support for SNMP on the switch, and how to configure the switch's SNMP agent.

Unless a particular version of SNMP is named, "SNMP" in this chapter refers to versions SNMPv1, SNMPv2c and SNMPv3.

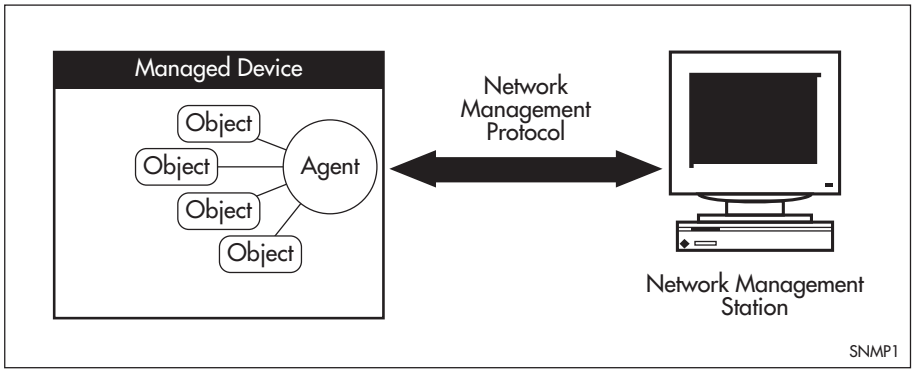
See also **Chapter 94, SNMP Commands** and **Chapter 95, SNMP MIBs**.

Network Management Framework

A network management system has the following components:

- One or more **managed devices**, each containing an agent that provides the management functions. A managed device may be any computing device with a network capability, for example, a host system, workstation, terminal server, printer, router, switch, bridge, hub or repeater.
- One or more **Network Management Stations (NMS)**. An NMS is a host system running a network management protocol and network management applications, enabling the user to manage the network.
- A **network management protocol** used by the NMS and agents to exchange information.

Figure 93-1: Components of a network management system



The Internet-standard Network Management Framework is the framework used for network management in the Internet. The framework was originally defined by the following documents:

- RFC 1155, *Structure and identification of management information for TCP/IP based internets* (referred to as the SMI), details the mechanisms used to describe and name the objects to be managed.
- RFC 1213, *Management Information Base for network management of TCP/ IP-based internets: MIB-II* (referred to as MIB-II), defines the core set of managed objects for the Internet suite of protocols. The set of managed objects can be extended by adding other MIBs specific to particular protocols, interfaces or network devices.
- RFC 1157, *A Simple Network Management Protocol (SNMP)*, is the protocol used for communication between management stations and managed devices.

Subsequent documents that have defined SNMPv2c are:

- RFC 1901, *Introduction to Community-based SNMPv2*
- RFC 1902, *Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1903, *Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1904, *Conformance Statements for Version 2 of the Simple Network Management Protocol*
- RFC 1905, *Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1906, *Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 1907, *Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)*
- RFC 2576, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*
- RFC 2578, *Structure of Management Information Version 2 (SMIv2)*
- RFC 2579, *Textual Conventions for SMIv2*
- RFC 2580, *Conformance Statements for SMIv2*

Subsequent documents that have defined SNMPv3 are:

- RFC 3410, *Introduction and Applicability Statements for Internet Standard Management Framework*
- RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*
- RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*
- RFC 3413, *Simple Network Management Protocol (SNMP) Applications*
- RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*
- RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
- RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*
- RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)*
- RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

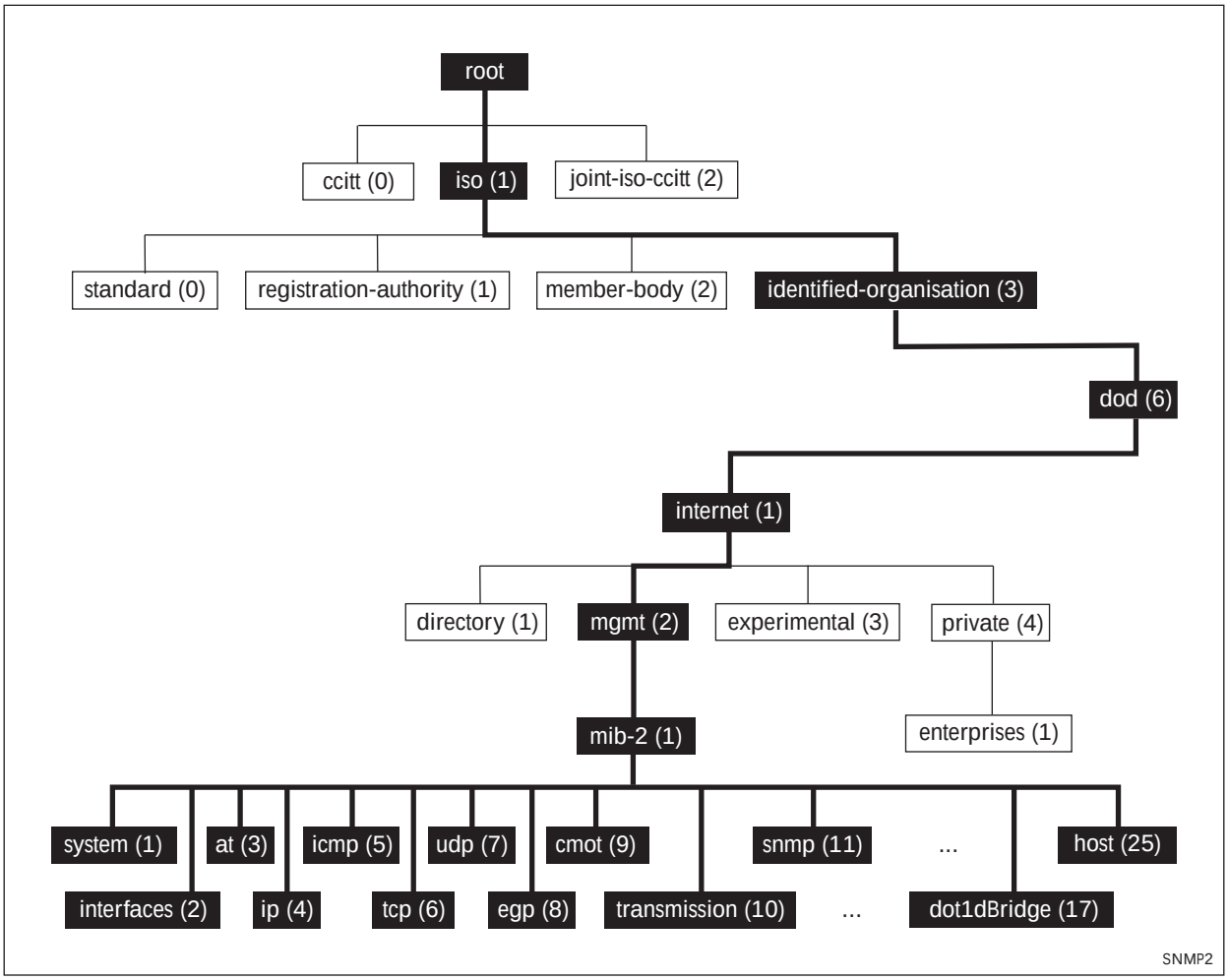
Structure of Management Information

The structure of management information (SMI) defines the schema for a collection of managed objects residing in a virtual store called the management information base (MIB). The information in a MIB includes administrative and operational configuration information, as well as counters of system events and activities.

The MIB is organized into a tree-like hierarchy in which nodes are each assigned an identifier consisting of a non-negative integer and an optional brief textual description.

Each managed object is represented by a leaf node and is defined by its name, syntax, access mode, status and description. It can also be specifically identified by its unique position within the tree. This position is expressed as a series of dot-delimited sub-identifiers that start at the root node and end in the sub-identifier at the particular object's leaf node. For example, in **Figure 93-2** the object named interfaces would be uniquely identified by the string of individual sub-identifiers, 1.3.6.1.2.1.2.

Figure 93-2: Top levels of the Internet-standard Management Information Base (MIB)



Objects defined in the Internet-standard MIB (MIB-II) reside in the mib(1) sub-tree.

Names

Names are used to identify managed objects, and are hierarchical in nature. An object identifier is a globally unique, authoritatively assigned sequence of non-negative integers which traverse the MIB tree from the root to the node containing the object.

Object identifiers may be represented in one of the following forms:

- Dotted notation lists the integer values found by traversing the tree from the root to the node in question, separated by dots. For example, the following identifies the MIB-II sub-tree:

  ```
  1.3.6.1.2.1
  ```

  The following identifies the sysDescr object in the system group of MIB-II:

  ```
  1.3.6.1.2.1.1.1
  ```

- Textual notation lists the textual descriptions found by traversing the tree from the root to the node in question, separated by spaces and enclosed in braces. The following example identifies the internet sub-tree:

  ```
  { iso org dod 1}
  ```

  The name may be abbreviated to a relative form. The following example identifies the first (directory) node of the internet sub-tree:

  ```
  { internet 1}
  ```

- Combined notation lists both the integer values and textual descriptions found by traversing the tree from the root to the node in question. The integer value is placed in parentheses after the textual description. The labels are separated by spaces and enclosed in braces. For example, the following identifies the first (directory) node in the internet sub-tree:

  ```
  {iso(1) org(3) dod(6) internet(1) 1}
  ```

  The name may be abbreviated to the following:

  ```
  directory(1)
  ```

Since there is no effective limit to the magnitude of non-negative integers, and no effective limit to the depth of the tree, the MIB provides an unlimited name space.

An object is also usually assigned an object descriptor. The object descriptor is a unique, mnemonic, printable string intended for humans to use when discussing the MIB.

Instances

Objects are just templates for data types. An actual value that can be manipulated by an NMS is an instance of an object. An instance is named by appending an instance identifier to the end of the object's object identifier. The instance identifier depends on the object's data type:

- If the object is not a column in a table, the instance identifier is 0 (zero). For example, the instance of the sysDescr object is:

  ```
  sysDescr.0
  or 1.3.6.1.2.1.1.1.0
  ```

- If the object is a column in a table, the method used to assign an instance identifier varies. Typically, the value of the index column or columns is used.

The object ifTable in MIB-II contains information about interfaces and is indexed by the interface number, ifIndex. The instance of the ifDescr object for the first interface is:

```
ifDescr.1
or 1.3.6.1.2.1.2.2.1.2.1
```

If the index column is an IP address, the entire IP address is used as the instance identifier. The object ipRouteTable in MIB-II contains information about IP routes and is indexed by the destination address, ipRouteDest. The instance of the ipRouteNextHop object for the route 131.203.9.0 is:

```
ipRouteNextHop.131.203.9.0
or 1.3.6.1.2.1.4.21.1.7.131.203.9.0
```

If the table has more than one index, the values of all the index columns are combined to form the instance identifier. The object tcpConnTable in MIB-II contains information about existing TCP connections and is indexed by the local IP address (tcpConnLocalAddress), the local port number (tcpConnLocalPort), the remote IP address (tcpConnRemAddress) and the remote port number (tcpConnRemPort) of the TCP connection. The instance of the tcpConnState object for the connection between 131.203.8.36,23 and 131.203.9.197,1066 is:

```
tcpConnState.131.203.8.36.23.131.203.9.197.1066
or 1.3.6.1.2.1.6.13.1.1.131.203.8.36.23.131.203.9.197.1066
```

Syntax

The syntax of an object describes the abstract data structure corresponding to that object type. For example, INTEGER or OCTET STRING.

Access

The access mode of an object describes the level of access for the object.

Access modes for MIB objects:

Access	Description
Read-only	The object's value can be read but not set.
Read-write	The object's value can be read and set.
Write-only	The object's value can be set but not read.
Not-accessible	The object's value cannot be read or set.

Status

The status of an object describes the implementation requirements for the object.

Status values for MIB objects:

Status	Description
Mandatory	Managed devices must implement the object.
Optional	Managed devices may implement the object.
Obsolete	Managed devices need no longer implement the object.
Deprecated	Managed devices should implement the object. However, the object may be deleted from the next version of the MIB. A new object with equal or superior functionality is defined.

Description

The definition of an object may include an optional textual description of the meaning and use of the object. This description is often essential for successful understanding of the object.

The SNMP Protocol

The SNMP protocol provides a mechanism for management entities, or stations, to extract information from the Management Information Base (MIB) of a managed device.

The normal method of accessing information in a MIB is to use a Network Management Station (NMS), typically a PC or workstation, to send commands to the managed device (in this case the switch) using the SNMP protocol.

SNMP can use a number of different protocols as its underlying transport mechanism, but the most common transport protocol, and the only one supported by the switch, is UDP. Therefore the IP module must be enabled and properly configured in order to use SNMP. SNMP trap messages are sent to UDP port 162; all other SNMP messages are sent to UDP port 161. The switch's SNMP agent accepts SNMP messages up to the maximum UDP length the switch can receive.

Other transport mappings have been defined (e.g. OSI [RFC 1418], AppleTalk [RFC 1419] and IPX [RFC 1420]), but the standard transport mapping for the Internet (and the one the switch uses) is UDP. The IP module must be enabled and configured correctly. See **Chapter 29, IP Addressing and Protocol Commands** for detailed descriptions of the commands required to enable and configure IP.

SNMP Versions

The switch supports SNMP version 1 (SNMPv1), SNMP version 2c (SNMPv2c) and SNMP Version 3 (SNMPv3). The three versions operate similarly.

SNMPv2c updated the original protocol, and offered the following main enhancements:

- a new format for trap messages.
- the get-bulk-request PDU allows for the retrieval of large amounts of data, including tables, with one message.
- more error codes mean that error responses to set messages have more detail than is possible with SNMPv1.
- three new exceptions to errors can be returned for get, get-next and get-bulk-request messages. These are: noSuchObject, noSuchInstance, and endOfMibView.

SNMPv3 provides significant enhancements to address the security weaknesses existing in the earlier versions. This is achieved by implementing two new major features:

- Authentication - by using password hashing and time stamping.
- Privacy - by using message encryption.

Support for multiple versions of SNMP is achieved by responding to each SNMP request with a response of the same version. For example, if an SNMPv1 request is sent to the switch, an SNMPv1 response is returned. If an SNMPv2c request is sent, an SNMPv2c response is returned. Therefore, authentication and encryption functions are not invoked when messages are detected as having either an SNMPv1 or SNMPv2c protocol format.

SNMP Messages

The SNMP protocol is termed simple because it has only six operations, or messages—get, get-next, get-response, set, and trap, and SNMPv2c also has the get-bulk-request message. The replies from the managed device are processed by the NMS and generally used to provide a graphical representation of the state of the network. The two major SNMP operations available to a management station for interacting with a client are the get and set operations. The SNMP set operator can lead to security breaches, since SNMP is not inherently very secure. When forced to operate in either SNMPv1 or v2 mode, when operating with older management stations for example, care must be taken in the choice and safe-guarding of community names, which are effectively passwords for SNMP.

Polling versus Event Notification

SNMP employs a polling paradigm. A Network Management Station (NMS) polls the managed device for information as and when it is required, by sending get-request, get-next-request, and/or get-bulk-request PDUs to the managed device. The managed device responds by returning the requested information in a get-response PDU. The NMS may manipulate objects in the managed device by sending a set-request PDU to the managed device.

The only time that a managed device initiates an exchange of information is in the special case of a trap PDU. A managed device may generate a limited set of traps to notify the NMS of critical events that may affect the ability of the NMS to communicate with the managed device or other managed devices on the network, and therefore to “manage” the network. Such events include the restarting or re-initialization of a device, a change in the status of a network link (up or down), or an authentication failure.

Message Format for SNMPv1 and SNMPv2c

Table 93-1: Fields in an SNMP message

Field	Function
Version	The version of the SNMP protocol. The value is version-1 (0) for the SNMP protocol as defined in RFC 1157, or version-2c (1) for the SNMP protocol as defined in RFC 1902.
Community	The name of an SNMP community, for authentication purposes
SNMP PDU	An SNMP Protocol Data Unit (PDU).

Table 93-2: SNMP PDUs

PDU	Function
get-request	Sent by an NMS to an agent, to retrieve the value of an object.
get-next-request	Sent by an NMS to an agent, to retrieve the value of the next object in the sub-tree. A sub-tree is traversed by issuing a get-request PDU followed by successive get-next-request PDUs.
get-bulk-request	Sent by an NMS to an agent to request a large amount of data with a single message. This is for SNMPv2c messages.
set-request	Sent by an NMS to an agent, to manipulate the value of an object. SNMP PDU Version Community
get-response	Sent by an agent to an NMS in response to a get-request, get-next-request, get-bulk-response, or set-request PDU.
trap	Sent by an agent to an NMS to notify the NMS of a extraordinary event.
report	Although not explicitly defined in the RFCs, reports are used for specific purposes such as EngineID discovery and time synchronization.

Table 93-3: Generic SNMP traps

Value	Meaning
coldStart	The agent is re-initializing itself. Objects may be altered.
warmStart	The agent is re-initializing itself. Objects are not altered.
linkDown	An interface has changed state from up to down.
linkUp	An interface has changed state from down to up.
authenticationFailure	An SNMP message has been received with an invalid community name.
egpNeighborLoss	An EGP peer has transitioned to down state.

SNMP Communities (Version v1 and v2c)

A community is a relationship between an NMS and an agent. The community name is used like a password for a trivial authentication scheme. Both SNMPv1 and SNMPv2c provide security based on the community name only. The concept of communities does not exist for SNMPv3, which instead provides for a far more secure communications method using entities, users, and groups.

Caution

We strongly recommend removing community membership from all SNMPv3 configured devices to prevent access to them via SNMPv1 and SNMv2c, which could bypass the additional SNMPv3 security features.

SNMPv3 Entities

Entities comprise one of the basic components of the SNMPv3 enhanced architecture. They define the functionality and internal structure of the SNMP managers and agents. An in-depth description of entities can be found in RFC 3411, on which the following text is based. SNMPv3 defines two entity types, a manager and an agent. Both entity types contain two basic components: an SNMP engine and a set of applications.

SNMP engine

The engine provides the basic services to support the agents component applications, in this respect it performs much of the functionality expected of the ISO Session and Presentation layers. These functions include message transmission and reception, authentication and encryption, and access control to its managed objects database (MIB). The SNMP engine comprises the following components:

- Dispatcher
- Message processing Subsystem
- Security Subsystem
- Access Control Subsystem

The only security subsystem presently supported is the user based security model (USM).

Each SNMP engine is identified by an snmpEngineID that must be unique within the management system. A one to one association exists between an engine and the entity that contains it.

Entity applications

The following applications are defined within the agent applications:

- Command Generator
- Notification Receiver
- Proxy Forwarder
- Command Responder
- Notification Originator
- Other

SNMPv3 Message Protocol Format

Table 93-4: SNMPv3 PDUs

Value	Meaning
msgVersion	Identifies the message format to be SNMPv3.
msgID	An identifier used between SNMP entities to coordinate message requests and responses. Note that a message response takes the msgID value of the initiating message.
msgMaxSize	Conveys the maximum message size (in octets) an integer between 484 and 2^{31}-1, supported by the sender of the message. Specified as msgFlags. A single octet whose last three bits indicate the operational mode for privacy, authentication, and report.
msgSecurityModel	An identifier used to indicate the security mode (i.e. SNMPv1, SNMPv2c or SNMPv3 to be used when processing the message. Note that although only the SNMPv3 identifier is accepted by the switch, these earlier version message formats are detected by the msgVersion field and processed appropriately.
msgAuthoritativeEngineID	The ID of the authoritative engine that relates to a particular message, i.e. the source engine ID for Traps, Responses and Reports, and the destination engine for Gets, GetNexts, Sets, and Informs.
msgAuthoritativeEngineBoots	A value that represents the number of times the authoritative engine has rebooted since its installation. Its value has the range 1 to 2^{31}-1.
msgAuthoritativeEngineTime	The number of seconds since the authoritative engine snmpEngineBoots counter was last incremented.
msgUserName	The name of the user (principal) on whose behalf the message is being exchanged.
msgAuthenticationParameters	If the message has been authenticated, this field contains a serialized OCTET STRING representing the first 12 octets of the HMAC-MD5-96 output done over the whole message.
msgPrivacyParameters	For encrypted data, this field contains the “salt” used to create the DES encryption Initialization Vector (IV).
ContextEngineID	Within a particular administrative domain, this field uniquely identifies an SNMP entity that may realize an instance of a context with a particular contextName
ContextName	A unique name given to a context within a particular SNMP entity.

SNMPv1 and SNMPv2c

Although software levels 2.6.3 and higher support the specific facilities of SNMP v1 and v2, their documentation is available to provide backward compatibility with older network management systems. The far superior security features offered by implementing SNMPv3 should be used wherever possible.

The switch's implementation of SNMPv1 is based on RFC 1157, *A Simple Network Management Protocol (SNMP)*, and RFC 1812, *Requirements for IP Version 4 Routers*.

When the SNMP agent is disabled, the agent does not respond to SNMP request messages. The agent is disabled by default. The current state and configuration of the SNMP agent can be displayed.

SNMP MIB Views for SNMPv1 and SNMPv2c

An SNMP MIB view is a arbitrary subset of objects in the MIB. Objects in the view may be from any part of the object name space, and not necessarily the same sub-tree. An SNMP community profile is the pairing of an SNMP access mode (read-only or read-write) with the access mode defined by the MIB for each object in the view. For each object in the view, the community profile defines the operations that can be performed on the object.

Pairing an SNMP community with an SNMP community profile determines the level of access that the agent affords to an NMS that is a member of the specified community. When an agent receives an SNMP message, it checks the community name encoded in the message. If the agent knows the community name, the message is deemed to be authentic and the sending SNMP entity is accepted as a member of the community. The community profile associated with the community name then determines the sender's view of the MIB and the operations that can be performed on objects in the view.

SNMP Communities

SNMP communities were introduced into SNMPv1 and retained in version 2c. Although the switch's software still supports communities, this is to provide backward compatibility with legacy management systems. Communities should not be used where a secure network is required. Instead, use the secure network features offered by SNMPv3.

An SNMP community is a pairing of an SNMP agent with a set of SNMP application entities. Communities are the main configuration item in the switch's implementation of SNMPv1 and v2, and are defined in terms of a list of IP addresses which define the SNMP application entities (trap hosts and management stations) in the community.

Important community names act as passwords and provide minimal authentication. Any SNMP application entity that knows a community name can read the value of any instance of any object in the MIB implemented in the switch. Any SNMP application entity that knows the name of a community with write access can change the value of any instance of any object in the MIB implemented in the switch, possibly affecting the operation of the switch. For this reason, take care with the security of community names.

When a trap is generated by the SNMP agent it is forwarded to all trap hosts in all communities. The community name and manager addresses are used to provide trivial authentication. An incoming SNMP message is deemed authentic if it contains a valid community name and originated from an IP address defined as a management station for that community.

When a community is disabled, the SNMP agent behaves as if the community does not exist and generates authentication failure traps for messages directed to the disabled community.

The SNMP agent does not support a default community called “public” with read-only access, traps disabled and open access as mandated in RFC 1812, as this is a security hole open for users who wish to use the switch with minimal modification to the default configuration. The default configuration of the switch has no defined communities. Communities must be explicitly created.

SNMP authentication (for SNMPv1 and v2) is a mechanism whereby an SNMP message is declared to be authentic, that is from an SNMP application entity actually in the community to which the message purports to belong. The mechanism may be trivial or secure. The only form of SNMP authentication implemented by the switch’s SNMP agent is trivial authentication. The authentication failure trap may be generated as a result of the failure to authentication an SNMP message.

Switch interfaces can be enabled or disabled via SNMP by setting the ifAdminStatus object in the ifTable of MIB-II MIB to ‘Up(1)’ or ‘Down(2)’ for the corresponding ifIndex. If it is not possible to change the status of a particular interface the switch returns an SNMP error message.

The switch’s implementation of the ifOperStatus object in the ifTable of MIB-II MIB supports two additional values—”Unknown(4)” and “Dormant(5)” (e.g. an inactive dial-on-demand interface).

Caution

An unauthorized person with knowledge of the appropriate SNMP community name could bring an interface up or down. Community names act as passwords for the SNMP protocol. When creating an SNMP community with write access, take care to select a secure community name and to ensure that only authorized personnel know it.

An SNMP MIB view is a subset of objects in the MIB that pertain to a particular network element. For example, the MIB view of a hub would be the objects relevant to management of the hub, and would not include IP routing table objects, for example. The switch’s SNMP agent does not allow the construction of MIB views. The switch supports all relevant objects from all MIBs that it implements.

Note that the switch’s standard set and show commands can also be used to access objects in the MIBs supported by the switch.

Defining management stations within communities

You can add management stations to a community either individually, by entering just its IP address, or you can enter a range of management stations by entering an IP address that ends with a ‘/’ character followed by a number between 1 and 32. The number that follows the ‘/’ character operates as an address mask to define a range of addresses for the management stations. The following example shows how to allocate a band of three binary addresses to a portion of the subnet 146.15.1.X

Example

In this example we make provision for up to 8 possible management stations within a community called “admin”.

Step 1:

Decide on the number of management stations that you want to assign to a particular subnet, then decide how many binary digits are required to define this number of addresses. In this case we need up to 8 management stations, so we will assign 3 binary digits (3 binary digits can provide 8 different values). To assign the last 3 binary digits for management stations, we assign a prefix that is a count of all binary digits in the address minus those to be assigned as management stations. In this case the prefix is 29; this being the number of binary digits in an IP address (32) minus the number of digits assigned to the management stations (3).

Step 2:

The method used in this step depends on whether or not the community already exists.

- If the community called “admin” does not exist, create a new community called “admin” and allocate a three binary digit block of addresses to the address subnet 146.15.1.X.
- If the community called “admin” already exists, allocate a three binary digit block of addresses to an existing community called “admin” with the address subnet 146.15.1.X.

For security reasons, the common management prefix should be larger than the IP subnet. This prevents stations on one subnet from being considered valid management stations on a different subnet.

Configuration Example (SNMPv1 and v2)

This example shows how to configure the switch’s SNMP agent. Two network management stations have been set up on a large network. The central NMS (IP address 192.168.11.5) monitors devices on the network and uses SNMP set messages to manage devices on the network. Trap messages are sent to this management station. The regional network management station (IP addresses 192.168.16.1) is used just to monitor devices on the network by using SNMP get messages. Link traps are enabled for all interfaces on this particular switch.

IP and VLANs must be correctly configured in order to access the SNMP agent in the switch. This is because the IP module handles both the TCP transport functions, and the UDP functions that enable datagrams to transport SNMP messages. See **Chapter 29, IP Addressing and Protocol Commands** for commands that enable and configure IP.

To configure SNMP

Step 1: Enable the SNMP agent.

Enable the SNMP agent and enable the generation of authenticate failure traps to monitor unauthorized SNMP access. SNMP is enabled by default in AlliedWare Plus.

```
awplus(config)# snmp-server enable trap auth
```

Step 2: Create a community with write access for the central NMS.

Create a write access community called “example1rw” for use by the central network management station at 192.168.11.5 Use an ACL to give the central NMS SNMP access to the switch using that community name.

```
awplus(config)# access-list 66 permit 192.168.11.5
awplus(config)# snmp-server community example1rw rw 66
```

Care must be taken with the security of community names. Do not use the names “private” or “public” in your network because they are too obvious. Community names act as passwords and provide only trivial authentication. Any SNMP application entity that knows a community name can read the value of any instance of any object in the MIB implemented in the switch. Any SNMP application entity that knows the name of a community with write access can change the value of any instance of any object in the MIB implemented in the switch, possibly affecting the operation of the switch.

SNMP V1 or V2c provide very minimal security. If security is a concern, you should use SNMPv3.

Step 3: Create a community with read-only access for the regional NMS.

Create a read-only access community called “example2ro” for use by the regional network management station at 192.168.16.1. Use an ACL to give the regional NMS SNMP access to the switch using that community name.

```
awplus(config)# access-list 67 permit 192.168.16.1
awplus(config)# snmp-server community example2ro ro 67
```

Step 4: Enable link traps.

Enable link traps for the desired interfaces. In this example, the NSMs are in VLAN 2 and VLAN 3 and other ports are in VLAN 1 for simplicity.

```
awplus(config)# interface vlan1-3
awplus(config-if)# snmp trap link-status
```

Note that link traps on VLANs are sent when the last port in the VLAN goes down. You will only see a trap for a VLAN if the trap host is in a different VLAN.

You can also enable link traps on channel groups and switch ports. For example, to enable traps on a range of switch ports:

```
awplus(config)# int port1.0.4-1.0.6
awplus(config-if)# snmp trap link-status
```

You can also enable link traps on channel groups and switch ports. For example, to enable traps on a range of switch ports:

Step 5: Configure trap hosts.

Specify the IP address or addresses that the traps will get sent to. In this example, traps will be sent to both NMSes.

```
awplus(config)# snmp-server host 192.168.11.5 version 2c
                example1rw
awplus(config)# snmp-server host 192.168.16.1 version 2c
                example2ro
```

Step 6: Check the configuration.

Check that the current configuration of the SNMP communities matches the desired configuration:

```
awplus# show snmp-server
awplus# show snmp-server community
awplus# show run snmp
```

This is the output of the **show snmp-server community** command for this example:

```
SNMP community information:
  Community Name ........... example1rw
    Access ................. Read-write
    View ................... none
  Community Name ........... example2ro
    Access ................. Read-only
    View ................... none
```

This is the output of the **show run snmp** command for this example:

```
no snmp-server ip
snmp-server enable trap auth
snmp-server community example1rw rw 66
snmp-server community example2ro 67
snmp-server host 192.168.1.2 version 2c example1rw
snmp-server host 192.168.2.2 version 2c example2ro
!
```

Check that the interface link up/down traps have been correctly configured:

```
awplus# show interface vlan1-3
```

This is the output of the **show interface** command for this example:

```
Interface vlan1
  Scope: both
  Link is UP, administrative state is UP
  Hardware is VLAN, address is 0009.41fd.c029
  index 201 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,RUNNING,MULTICAST>
  SNMP link-status traps: Sending (suppressed after 20 traps in 60 sec)
  Bandwidth 1g
    input packets 4061, bytes 277043, dropped 0, multicast   packets 3690
    output packets 190, bytes 18123, multicast packets 0 broadcast packets 0
Interface vlan2
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0009.41fd.c029
  IPv4 address 192.168.11.50/24 broadcast 192.168.11.255
  index 202 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  SNMP link-status traps: Sending (suppressed after 20 traps in 60 sec)
  Bandwidth 1g
    input packets 568, bytes 42309, dropped 0, multicast packets 0
    output packets 183, bytes 18078, multicast packets 0 broadcast packets 0
Interface vlan3
  Scope: both
  Link is DOWN, administrative state is UP
  Hardware is VLAN, address is 0009.41fd.c029
  IPv4 address 192.168.16.50/24 broadcast 192.168.16.255
  index 203 metric 1 mtu 1500
  arp ageing timeout 300
  <UP,BROADCAST,MULTICAST>
  SNMP link-status traps: Sending (suppressed after 20 traps in 60 sec)
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast packets 0
```

SNMPv3

SNMPv3 is the third version of the Simple Network Management Protocol. The architecture comprises the following:

- entities that may be either managers, agents, or both
- a management information base (MIB)
- a transport protocol

At least one manager node runs the SNMP management software in every configuration. Managed devices such as routers, servers, and workstations are equipped with an agent software module. The agent provides access to local objects in the MIB that reflect activity and resources at the node. The agent also responds to manager commands to retrieve values from, and set values in the MIB.

SNMP MIB Views for SNMPv3

An SNMP MIB view is a arbitrary subset of objects in the MIB. Objects in the view may be from any part of the object name space, and not necessarily the same sub-tree.

SNMP Groups

Groups were introduced as part of SNMPv3. They are the means by which users are assigned their views and access control policy. Once a group has been created, users can be added to them. In practice a number of groups would be created, each with varying views and access security requirements. Users would then be added to their most appropriate groups. Each Group name and Security Level pair must be unique within a switch.

SNMP Users

Users were introduced as part of SNMPv3. From a system perspective a user is represented as an entity stored in a table that defines the access and authentication criteria to be applied to access or modify the SNMP MIB data.

Note SNMP **Target Addresses** and **Target Params** in SNMPv3 are not currently supported in Alliedware Plus software.

Configuration Example (SNMPv3)

This example shows how to configure the switch's SNMP agent. Two network management stations have been set up on a large network. The central NMS (IP address 192.168.11.5) monitors devices on the network and uses SNMP set messages to manage devices on the network. Trap messages are sent to this management station.

The IP module must be enabled and correctly configured in order to access the SNMP agent in the switch, since the IP module handles the UDP datagrams used to transport SNMP messages.

To configure SNMP

Step 1: Enable the SNMP agent.

Enable the SNMP agent and enable the generation of authenticate failure traps to monitor unauthorized SNMP access. SNMP is enabled by default in AlliedWare Plus.

Step 2: Add SNMP views.

You can specify views using their OID or the predefined MIB name.

```
awplus(config)# snmp-server view atmib 1.3.6.1.2.14
included

awplus(config)# snmp-server view atmib alliedtelesis
included
```

Step 3: Add SNMP group.

```
awplus(config)# snmp-server group ord-user noauth read
atmib

awplus(config)# snmp-server group admin-user auth read
atmib write atmib notify atmi
```

Step 4: Add SNMP users.

Add users to the groups by using commands such as:

```
awplus(config)# snmp-server user ken admin-user auth md5
mercury
```

Using SNMP to Manage Files and Software

The Allied Telesis Enterprise MIB (**Chapter 95, SNMP MIBs**) includes objects for managing files and software on the switch. This section includes procedures for using MIB objects on the switch to perform some common tasks, via an SNMP management application:

- **"Copy a File to or from a TFTP Server" on page 93.20**
- **"Upgrade Software and Configuration Files" on page 93.22**

For more details about the Allied Telesis Enterprise MIB and public MIBs on the switch, see **Chapter 95, SNMP MIBs**.

Copy a File to or from a TFTP Server

Use this procedure to copy a file (for example, a software version file) to the switch from a TFTP server, or to copy a file (for example, a configuration file) from the switch to a TFTP server. The MIB objects in this procedure reside in the module atFilev2 { modules 600 }, with object ID 1.3.6.1.4.1.207.8.4.4.4.600. For detailed descriptions of the MIB objects used in this procedure, and other file management MIB objects, see **"AT-FILEv2-MIB" on page 95.31**. Other MIB objects can be used in a similar way for moving and deleting files on the switch.

Table 93-5: Procedure for copying a file to or from a device using a TFTP server

	Do this ...	By setting or reading this MIB object ...	Whose object ID is ...	To this value ...
1.	If the source device is part of a stack, set the stack ID. For a standalone switch, keep the default value, 1.	atFilev2SourceStackId	{ atFilev2Operation 1 }	*<stack-id>*
2.	If the destination device is part of a stack, set the stack ID.	atFilev2DestinationStackId	{ atFilev2Operation 4 }	*<stack-id>*
3.	Set the source device.	atFilev2SourceDevice	{ atFilev2Operation 2 }	4 (TFTP) or 1 (Flash)
4.	Set the destination device.	atFilev2DestinationDevice	{ atFilev2Operation 5 }	4 (TFTP) or 1 (Flash)
5.	Set the source filename. Include the path (if any) but not the device.	atFilev2SourceFileName	{ atFilev2Operation 3 }	*<source-filename>* e.g. /awp/config/admin.cfg
6.	Set the destination filename. Include the path (if any) but not the device.	atFilev2DestinationFileName	{ atFilev2Operation 6 }	*<dest-filename>* e.g. /config/admin.cfg

Table 93-5: Procedure for copying a file to or from a device using a TFTP server

	Do this ...	By setting or reading this MIB object ...	Whose object ID is ...	To this value ...
7.	Set the IP address of the TFTP server.	atFilev2TftpIPAddr	{ atFilev2Tftp_4 1 }	*<ip-addr>*
8.	Check that no other transfer is in progress, and that the required parameters have been set.	atFilev2CopyBegin	{ atFilev2Operation 7 }	Read: idle
9.	Start the file transfer.	atFilev2CopyBegin	{ atFilev2Operation 7 }	Set: 1
10	Monitor file transfer progress.	atFilev2CopyBegin	{ atFilev2Operation 7 }	Read: In progress: copying *<src>* --> *<dst>* *or* Success: copy *<src>* --> *<dst>* success *or* Failure: copy *<src>* --> *<dst>* failure: *<err-msg>*

Upgrade Software and Configuration Files

Use this procedure to upgrade to a new software version and boot configuration file. For detailed descriptions of the MIB objects used in this procedure, and other MIB objects for managing software installation and configuration files, see **"AT-SETUP-MIB" on page 95.55**.

Table 93-6: Procedure for upgrading to a new software version and boot configuration

	Do this ...	By reading or setting this MIB object ...	Whose object ID is ...	To this value ...
1.	Check that you have enough flash memory for the currently running software file, the new software version file, and any configuration scripts required.			
2.	Check the version and name of the software currently running.	currSoftVersion currSoftName	1.3.6.1.4.1.207.8.4.4.4.500.2.1.1 1.3.6.1.4.1.207.8.4.4.4.500.2.1.2	Read: <*software-name*> <*software-version*>
3.	If you do not already have the currently running software as a software version file in flash, save the currently running software with a file name to the flash root.	currSoftSaveToFile	1.3.6.1.4.1.207.8.4.4.4.500.2.1.4	Set: <*backup-filename.rel*>
4.	Check that the file saved successfully. (The most common failures result from lack of flash memory space.)	currSoftSaveStatus	1.3.6.1.4.1.207.8.4.4.4.500.2.1.5	Read: ■ 1 (idle) - there is no release file save operation in progress ■ 2 (success) - the last release file save operation completed successfully ■ 3 (failure) - the last release file save operation failed ■ 4 (saving) - a release file save operation is currently in progress
5.	Copy the new software version file to flash memory on the device	See **Table 93-5**.		
6.	Set the new release file to be the current release that the device will install and run the next time it restarts. Include the path.	nextBootPath	1.3.6.1.4.1.207.8.4.4.4.500.2.2.2	Set: <*next-filename*> e.g.: `flash:/release.rel`
7.	Check the version of release file set to install next.	nextBootVersion	1.3.6.1.4.1.207.8.4.4.4.500.2.2.1	Read: <*software-version*>

Table 93-6: Procedure for upgrading to a new software version and boot configuration(cont.)

	Do this ...	By reading or setting this MIB object ...	Whose object ID is ...	To this value ...
8.	Set the previous release file to be the backup release that the device will install and run if the device fails to boot successfully with the new release file. Include the path.	bckpPath	1.3.6.1.4.1.207.8.4.4.4.500.2.3.2	Set: *<backup-filename>* e.g.: `flash:/release.rel`
9.	Check the version of backup release file.	bckpVersion	1.3.6.1.4.1.207.8.4.4.4.500.2.3.1	Read: *<software-version>*
10.	If necessary, copy a configuration file to the device (**Table 93-5**), or save the current running configuration to a file in the root directory of flash. To save the running configuration, specify the filename, but not a device or path.	See **Table 93-5**. or runCnfgSaveAs	1.3.6.1.4.1.207.8.4.4.4.500.3.1.1	Set: *<filename.cfg>* e.g.: myconfig.cfg
11.	Check and if necessary set the file the device will use for configuration when it restarts. Include the full path.	bootCnfgPath	1.3.6.1.4.1.207.8.4.4.4.500.3.2.1	Read/set: *<filename.cfg>* e.g.: flash:/myconfig.cfg
12.	Check that a boot configuration file matching the boot configuration path exists.	bootCnfgExists	1.3.6.1.4.1.207.8.4.4.4.500.3.2.2	Read: TRUE (1) or FALSE (2)
13.	Check that the default configuration file flash:/default.cfg exists.	dfltCnfgExists	1.3.6.1.4.1.207.8.4.4.4.500.3.3.2	Read: TRUE (1) or FALSE (2)
14.	Restart the device.	restartDevice	1.3.6.1.4.1.207.8.4.4.4.500.1	1

Chapter 94: SNMP Commands

Command List 94.2
debug snmp 94.2
show counter snmp-server 94.4
show debugging snmp 94.7
show running-config snmp 94.7
show snmp-server 94.8
show snmp-server community 94.8
show snmp-server group 94.9
show snmp-server user 94.9
show snmp-server view 94.10
snmp trap link-status 94.11
snmp trap link-status suppress 94.12
snmp-server 94.14
snmp-server community 94.16
snmp-server contact 94.17
snmp-server enable trap 94.18
snmp-server engineID local 94.21
snmp-server engineID local reset 94.23
snmp-server group 94.24
snmp-server host 94.25
snmp-server location 94.27
snmp-server source-interface 94.28
snmp-server startup-trap-delay 94.29
snmp-server user 94.30
snmp-server view 94.32
undebug snmp 94.33

Command List

This chapter provides an alphabetical reference for commands used to configure SNMP. For more information, see **Chapter 93, SNMP Introduction**, and **Chapter 95, SNMP MIBs**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

debug snmp

This command enables SNMP debugging.

The **no** variant of this command disables SNMP debugging.

Syntax

```
debug snmp [all|detail|error-string|process|receive|send|xdump]
no debug snmp [all|detail|error-string|process|receive|send|xdump]
```

Parameter	Description
`all`	Enable or disable the display of all SNMP debugging information.
`detail`	Enable or disable the display of detailed SNMP debugging information.
`error-string`	Enable or disable the display of debugging information for SNMP error strings.
`process`	Enable or disable the display of debugging information for processed SNMP packets.
`receive`	Enable or disable the display of debugging information for received SNMP packets.
`send`	Enable or disable the display of debugging information for sent SNMP packets.
`xdump`	Enable or disable the display of hexadecimal dump debugging information for SNMP packets.

Mode Privileged Exec and Global Configuration

Example To start SNMP debugging, use the command:

```
awplus# debug snmp
```

To start SNMP debugging, showing detailed SNMP debugging information, use the command:

```
awplus# debug snmp detail
```

To start SNMP debugging, showing all SNMP debugging information, use the command:

```
awplus# debug snmp all
```

Related Commands

show debugging snmp
terminal monitor
undebug snmp

show counter snmp-server

This command displays counters for SNMP messages received by the SNMP agent.

Syntax `show counter snmp-server`

Mode User Exec and Privileged Exec

Example To display the counters for the SNMP agent, use the command:

```
awplus# show counter snmp-server
```

Output **Figure 94-1: Example output from the show counter snmp-server command**

```
SNMP-SERVER counters
inPkts                  ......... 11
inBadVersions           ......... 0
inBadCommunityNames     ......... 0
inBadCommunityUses      ......... 0
inASNParseErrs          ......... 0
inTooBigs               ......... 0
inNoSuchNames           ......... 0
inBadValues             ......... 0
inReadOnlys             ......... 0
inGenErrs               ......... 0
inTotalReqVars          ......... 9
inTotalSetVars          ......... 0
inGetRequests           ......... 2
inGetNexts              ......... 9
inSetRequests           ......... 0
inGetResponses          ......... 0
inTraps                 ......... 0
outPkts                 ......... 11
outTooBigs              ......... 0
outNoSuchNames          ......... 2
outBadValues            ......... 0
outGenErrs              ......... 0
outGetRequests          ......... 0
outGetNexts             ......... 0
outSetRequests          ......... 0
outGetResponses         ......... 11
outTraps                ......... 0
UnSupportedSecLevels    ......... 0
NotInTimeWindows        ......... 0
UnknownUserNames        ......... 0
UnknownEngineIDs        ......... 0
WrongDigest             ......... 0
DecryptionErrors        ......... 0
UnknownSecModels        ......... 0
InvalidMsgs             ......... 0
UnknownPDUHandlers      ......... 0
```

Table 94-1: Parameters in the output of the show counter snmp-server command

Parameter	Meaning
`inPkts`	The total number of SNMP messages received by the SNMP agent.
`inBadVersions`	The number of messages received by the SNMP agent for an unsupported SNMP version. It drops these messages.The SNMP agent on your device supports versions 1, 2C, and 3.
`inBadCommunityNames`	The number of messages received by the SNMP agent with an unrecognized SNMP community name. It drops these messages.

Table 94-1: Parameters in the output of the show counter snmp-server command

Parameter	Meaning
inBadCommunityUses	The number of messages received by the SNMP agent where the requested SNMP operation is not permitted from SNMP managers using the SNMP community named in the message.
inASNParseErrs	The number of ASN.1 or BER errors that the SNMP agent has encountered when decoding received SNMP Messages.
inTooBigs	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'tooBig'. This is sent by an SNMP manager to indicate that an exception occurred when processing a request from the agent.
inNoSuchNames	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'noSuchName'. This is sent by an SNMP manager to indicate that an exception occurred when processing a request from the agent.
inBadValues	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'badValue'. This is sent by an SNMP manager to indicate that an exception occurred when processing a request from the agent.
inReadOnlys	The number of valid SNMP PDUs received by the SNMP agent where the value of the error-status field is 'readOnly'. The SNMP manager should not generate a PDU which contains the value 'readOnly' in the error-status field. This indicates that there is an incorrect implementations of the SNMP.
inGenErrs	The number of SNMP PDUs received by the SNMP agent where the value of the error-status field is 'genErr'.
inTotalReqVars	The number of MIB objects that the SNMP agent has successfully retrieved after receiving valid SNMP Get-Request and Get-Next PDUs.
inTotalSetVars	The number of MIB objects that the SNMP agent has successfully altered after receiving valid SNMP Set-Request PDUs.
inGetRequests	The number of SNMP Get-Request PDUs that the SNMP agent has accepted and processed.
inGetNexts	The number of SNMP Get-Next PDUs that the SNMP agent has accepted and processed.
inSetRequests	The number of SNMP Set-Request PDUs that the SNMP agent has accepted and processed.
inGetResponses	The number of SNMP Get-Response PDUs that the SNMP agent has accepted and processed.
inTraps	The number of SNMP Trap PDUs that the SNMP agent has accepted and processed.
outPkts	The number of SNMP Messages that the SNMP agent has sent.
outTooBigs	The number of SNMP PDUs that the SNMP agent has generated with the value 'tooBig' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.

Table 94-1: Parameters in the output of the show counter snmp-server command

Parameter	Meaning
outNoSuchNames	The number of SNMP PDUs that the SNMP agent has generated with the value `noSuchName' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
outBadValues	The number of SNMP PDUs that the SNMP agent has generated with the value 'badValue' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
outGenErrs	The number of SNMP PDUs that the SNMP agent has generated with the value 'genErr' in the error-status field. This is sent to the SNMP manager to indicate that an exception occurred when processing a request from the manager.
outGetRequests	The number of SNMP Get-Request PDUs that the SNMP agent has generated.
outGetNexts	The number of SNMP Get-Next PDUs that the SNMP agent has generated.
outSetRequests	The number of SNMP Set-Request PDUs that the SNMP agent has generated.
outGetResponses	The number of SNMP Get-Response PDUs that the SNMP agent has generated.
outTraps	The number of SNMP Trap PDUs that the SNMP agent has generated.
UnSupportedSecLevels	The number of received packets that the SNMP agent has dropped because they requested a securityLevel unknown or not available to the SNMP agent.
NotInTimeWindows	The number of received packets that the SNMP agent has dropped because they appeared outside of the authoritative SNMP agent's window.
UnknownUserNames	The number of received packets that the SNMP agent has dropped because they referenced an unknown user.
UnknownEngineIDs	The number of received packets that the SNMP agent has dropped because they referenced an unknown snmpEngineID.
WrongDigest	The number of received packets that the SNMP agent has dropped because they didn't contain the expected digest value.
DecryptionErrors	The number of received packets that the SNMP agent has dropped because they could not be decrypted.
UnknownSecModels	The number of messages received that contain a security model that is not supported by the server. Valid for SNMPv3 messages only.
InvalidMsgs	The number of messages received where the security model is supported but the authentication fails. Valid for SNMPv3 messages only.
UnknownPDUHandlers	The number of times the SNMP handler has failed to process a PDU. This is a system debugging counter.

Related Commands show snmp-server

show debugging snmp

This command displays whether SNMP debugging is enabled or disabled.

Syntax `show debugging snmp`

Mode User Exec and Privileged Exec

Example To display the status of SNMP debugging, use the command:

```
awplus# show debugging snmp
```

Output **Figure 94-2: Example output from the show debugging snmp command**

```
Snmp (SMUX) debugging status:
  Snmp debugging is on
```

Related Commands debug snmp

show running-config snmp

This command displays the current configuration of SNMP on your device.

Syntax `show running-config snmp`

Mode Privileged Exec

Example To display the current configuration of SNMP on your device, use the command:

```
awplus# show running-config snmp
```

Output **Figure 94-3: Example output from the show running-config snmp command**

```
snmp-server contact AlliedTelesis
snmp-server location Philippines
snmp-server group grou1 auth read view1 write view1 notify view1
snmp-server view view1 1 included
snmp-server community public
snmp-server user user1 group1 auth md5 password priv des
password
```

Related Commands show snmp-server

show snmp-server

This command displays the status and current configuration of the SNMP server.

Syntax `show snmp-server`

Mode Privileged Exec

Example To display the status of the SNMP server, use the command:

```
awplus# show snmp-server
```

Output **Figure 94-4: Example output from the show snmp-server command**

```
SNMP Server ......................... Enabled
IP Protocol ......................... IPv4
SNMPv3 Engine ID (configured name) ... Not set
SNMPv3 Engine ID (actual) ............ 0x80001f888021338e4747b8e607
```

Related Commands
debug snmp
show counter snmp-server
snmp-server
snmp-server engineID local
snmp-server engineID local reset

show snmp-server community

This command displays the SNMP server communities configured on the device. SNMP communities are specific to v1 and v2c.

Syntax `show snmp-server community`

Mode Privileged Exec

Example To display the SNMP server communities, use the command:

```
awplus# show snmp-server community
```

Output **Figure 94-5: Example output from the show snmp-server community command**

```
SNMP community information:
  Community Name ........... public
    Access ................. Read-only
    View ................... none
```

Related Commands
show snmp-server
snmp-server community

show snmp-server group

This command displays information about SNMP server groups. This command is used with SNMP version 3 only.

Syntax `show snmp-server group`

Mode Privileged Exec

Example To display the SNMP groups configured on the device, use the command:

```
awplus# show snmp-server group
```

Output **Figure 94-6: Example output from the show snmp-server group command**

```
SNMP group information:
  Group name ..... guireadgroup
    Security Level ..... priv
    Read View ..... guiview
    Write View ..... none
    Notify View ..... none

  Group name ..... guiwritegroup
    Security Level ..... priv
    Read View ..... none
    Write View ..... guiview
    Notify View ..... none
```

Related Commands **show snmp-server**
snmp-server group

show snmp-server user

This command displays the SNMP server users and is used with SNMP version 3 only.

Syntax `show snmp-server user`

Mode Privileged Exec

Example To display the SNMP server users configured on the device, use the command:

```
awplus# show snmp-server user
```

Output **Figure 94-7: Example output from the show snmp-server user command**

```
Name                Group name          Auth        Privacy
-------             -------------       -------     ----------
freddy              guireadgroup        none        none
```

Related Commands **show snmp-server**
snmp-server user

show snmp-server view

This command displays the SNMP server views and is used with SNMP version 3 only.

Syntax `show snmp-server view`

Mode Privileged Exec

Example To display the SNMP server views configured on the device, use the command:

```
awplus# show snmp-server view
```

Output **Figure 94-8: Example output from the show snmp-server view command**

```
SNMP view information:
  View Name ..............  view1
    OID ..................  1
    Type ..................  included
```

Related Commands **show snmp-server**
snmp-server view

snmp trap link-status

Use this command to enable SNMP to send link status notifications (traps) for the interfaces when an interface goes up (linkUp) or down (linkDown).

Use the **no** variant of this command to disable the sending of link status notifications.

Syntax

```
snmp trap link-status
no snmp trap link-status
```

Default By default, link status notifications are disabled.

Mode Interface Configuration

Usage The link status notifications can be enabled for the following interface types:

- switch port (e.g. port 1.0.1)
- VLAN (e.g. vlan2)
- static and dynamic link aggregation (e.g. sa2, po2)

To specify where notifications are sent, use the **snmp-server host** command on page 94.25. To configure the switch globally to send other notifications, use the **snmp-server enable trap** command on page 94.18.

Examples To enable SNMP to send link status notifications for ports 1.0.2 to 1.0.6, use following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-1.0.6
awplus(config-if)# snmp trap link-status
```

To disable the sending of link status notifications for port 1.0.2, use following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no snmp trap link-status
```

Related Commands

show interface
snmp trap link-status suppress
snmp-server enable trap
snmp-server host

snmp trap link-status suppress

Use this command to enable the suppression of link status notifications (traps) for the interfaces beyond the specified threshold, in the specified interval.

Use the **no** variant of this command to disable the suppression of link status notifications for the ports.

Syntax

```
snmp trap link-status suppress {time {<1-60>|default}|
   threshold {<1-20>|default}}

no snmp trap link-status suppress
```

Parameter	Description
`time`	Set the suppression timer for link status notifications.
<1-60>	The suppress time in seconds.
`default`	The default suppress time in seconds (60).
`threshold`	Set the suppression threshold for link status notifications. This is the number of link status notifications after which to suppress further notifications within the suppression timer interval.
<1-20>	The number of link status notifications.
`default`	The default number of link status notifications (20).

Default By default, if link status notifications are enabled (they are enabled by default), the suppression of link status notifications is enabled: notifications that exceed the notification threshold (default 20) within the notification timer interval (default 60 seconds) are not sent.

Mode Interface Configuration

Usage An unstable network can generate many link status notifications. When notification suppression is enabled, a suppression timer is started when the first link status notification of a particular type (linkUp or linkDown) is sent for an interface. If the threshold number of notifications of this type is sent before the timer reaches the suppress time, any further notifications of this type generated for the interface during the interval are not sent. At the end of the interval, the sending of link status notifications resumes, until the threshold is reached in the next interval.

Examples To enable the suppression of link status notifications for ports 1.0.2 to 1.0.6 after 10 notifications have been sent in 40 seconds, use following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2-1.0.6
awplus(config-if)# snmp trap link-status suppress time 40
                   threshold 10
```

To disable the suppression link status notifications for port 1.0.2, use following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no snmp trap link-status suppress
```

Related Commands **show interface**
snmp trap link-status

snmp-server

Use this command to enable the SNMP agent (server) on the switch. The SNMP agent receives and processes SNMP packets sent to the switch, and generates notifications (traps) that have been enabled by the **snmp-server enable trap** command on page 94.18.

Use the **no** variant of this command to disable the SNMP agent on the switch. When SNMP is disabled, SNMP packets received by the switch are discarded, and no notifications are generated. This does not remove any existing SNMP configuration.

Syntax

```
snmp-server [ip|ipv6]
no snmp-server [ip|ipv6]
```

Parameter	Description
ip	Enable or disable the SNMP agent for IPv4.
ipv6	Enable or disable the SNMP agent for IPv6.

Default By default, the SNMP agent is enabled for both IPv4 and IPv6. If neither the **ip** parameter nor the **ipv6** parameter is specified for this command, then SNMP is enabled or disabled for both IPv4 and IPv6.

Mode Global Configuration

Examples To enable SNMP on the switch for both IPv4 and IPv6, use the commands:

```
awplus# configure terminal
awplus(config)# snmp-server
```

To enable the SNMP agent for IPv4 on the device, use the commands:

```
awplus# configure terminal
awplus(config)# snmp-server ip
```

To disable the SNMP agent for both IPv4 and IPv6 on the switch, use the commands:

```
awplus# configure terminal
awplus(config)# no snmp-server
```

To disable the SNMP agent for IPv4, use the commands:

```
awplus(config)# no snmp-server ipv4
```

Related Commands

show snmp-server
show snmp-server community
show snmp-server user
snmp-server community
snmp-server contact
snmp-server enable trap
snmp-server engineID local
snmp-server group

snmp-server host
snmp-server location
snmp-server view

snmp-server community

This command creates an SNMP community, optionally setting the access mode for the community. The default access mode is read only. If view is not specified, the community allows access to all the MIB objects. The SNMP communities are only valid for SNMPv1 and v2c and provide very limited security. Communities should not be used when operating SNMPv3.

The **no** variant of this command removes an SNMP community. The specified community must already exist on the device.

Syntax

```
snmp-server community <community-name> {view <view-name>|ro|rw|
    <access-list>}
no snmp-server community <community-name> [{view <view-name>|<access-
    list>}]
```

Parameter	Description
`<community-name>`	Community name. The community name is a case sensitive string of up to 20 characters.
`view`	Configure SNMP view. If view is not specified, the community allows access to all the MIB objects.
`<view-name>`	View name. The view name is a string up to 20 characters long and is case sensitive.
`ro`	Read-only community.
`rw`	Read-write community.
`<access-list>`	*<1-99>* Access list number.

Mode Global Configuration

Example The following command creates an SNMP community called “public” with read only access to all MIB variables from any management station.

```
awplus# configure terminal
awplus(config)# snmp-server community public ro
```

The following command removes an SNMP community called “public”

```
awplus# configure terminal
awplus(config)# no snmp-server community public
```

Related Commands
show snmp-server
show snmp-server community
snmp-server view

snmp-server contact

This command sets the contact information for the system. The contact name is:

- displayed in the output of the **show system** command
- stored in the MIB object sysContact

The **no** variant of this command removes the contact information from the system.

Syntax

```
snmp-server contact <contact-info>
no snmp-server contact
```

Parameter	Description
<contact-info>	The contact information for the system, from 0 to 255 characters long. Valid characters are any printable character and spaces.

Mode Global Configuration

Example To set the system contact information to "support@alliedtelesis.co.nz", use the command:

```
awplus# configure terminal
awplus(config)# snmp-server contact support@alliedtelesis.co.nz
```

Related Commands

show system
snmp-server location
snmp-server group

snmp-server enable trap

Use this command to enable the switch to send the specified notifications (traps).

Note that the Environmental Monitoring traps are enabled by default. So you do not need to issue this command for the Environmental Monitoring traps since these are enabled by default. SNMP environmental monitoring traps defined in AT-ENVMONv2-MIB are enabled by default.

Use the **no** variant of this command to disable the sending of the specified notifications.

Syntax

```
snmp-server enable trap {[atmf][atmflink][atmfnode][atmfrr][auth]
    [dhcpsnooping] [epsr] [lldp] [loopprot] [mstp] [nsm] [ospf] [pim]
    [power-inline] [rmon] [thrash-limit] [vcs] [vrrp]}

no snmp-server enable trap {[atmf][atmflink][atmfnode][atmfrr][auth]
    [dhcpsnooping] [epsr] [lldp] [loopprot] [mstp] [nsm] [ospf] [pim]
    [power-inline] [rmon] [thrash-limit] [vcs] [vrrp]}
```

Parameter	Description
`atmf`	ATMF traps.
`atmflink`	ATMF Link traps.
`atmfnode`	ATMF Node traps.
`atmfrr`	ATMF Reboot Rolling traps.
`auth`	Authentication failure.
`dhcpsnooping`	DHCP snooping and ARP security traps. These notifications must also be set using the **ip dhcp snooping violation** command on page 80.21, and/or the **arp security violation** command on page 80.3.
`epsr`	EPSR traps.
`lldp`	Link Layer Discovery Protocol (LLDP) traps. These notifications must also be enabled using the **lldp notifications** command on page 97.13, and/or the **lldp med-notifications** command on page 97.8.
`loopprot`	Loop Protection traps.
`mstp`	MSTP traps.
`nsm`	NSM traps.
`ospf`	OSPF traps.
`pim`	PIM traps.
`power-inline`	Power-inline traps (Power Ethernet MIB RFC 3621).
`rmon`	RMON traps.
`thrash-limit`	MAC address Thrash Limiting traps.
`vcs`	VCS traps.
`vrrp`	Virtual Router Redundancy (VRRP) traps.

Default By default, no notifications are generated.

Mode Global Configuration

Usage This command cannot be used to enable link status notifications globally. To enable link status notifications for particular interfaces, use the **snmp trap link-status** command.

To specify where notifications are sent, use the **snmp-server host** command.

Note that more than one trap can be configured with one command entry, and also note this command applied to notifications send by SNMP version 3.

Examples To enable the device to send a notification if an atmf node changes its status, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap atmfnode
```

To enable the device to send PoE related traps, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap power-inline
```

To disable PoE traps being sent out by the switch, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server enable power-inline
```

To enable the device to send MAC address Thrash Limiting traps, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap thrash-limit
```

To disable the device from sending MAC address Thrash Limiting traps, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server enable trap thrash-limit
```

To enable the device to send OSPF and VRRP-related traps, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server enable trap ospf vrrp
```

To disable OSPF traps being sent out by the switch, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server enable trap ospf
```

Related Commands

show snmp-server
show ip dhcp snooping
snmp trap link-status
snmp-server host

snmp-server engineID local

Use this command to configure the SNMPv3 engine ID. The SNMPv3 engine ID is used to uniquely identify the SNMPv3 agent on a switch when communicating with SNMP management clients. Once an SNMPv3 engine ID is assigned, this engine ID is permanently associated with the switch until you change it.

Use the **no** variant of this command to set the user defined SNMPv3 engine ID to a system generated pseudo-random value by resetting the SNMPv3 engine. The **no snmp-server engineID local** command has the same effect as the **snmp-server engineID local default** command. Note that the **snmp-server engineID local reset** command is used to force the system to generate a new engine ID when the current engine ID is also system generated.

Syntax

```
snmp-server engineID local {<engine-id>|default}
no snmp-server engineID local
```

Parameter	Description
<engine-id>	Specify SNMPv3 Engine ID value, a string of up to 27 characters.
`default`	Set SNMPv3 engine ID to a system generated value by resetting the SNMPv3 engine, provided the current engine ID is user defined. If the current engine ID is system generated, use the **snmp-server engineID local reset** command to force the system to generate a new engine ID.

Mode Global Configuration

Usage All switches must have a unique engine ID which is permanently set unless it is configured by the user.

In a stacked environment, if the same engine ID was automatically generated for all members of the stack, conflicts would occur if the stack was dismantled. Therefore, each member of the stack will generate its own engine ID and the stack master's ID is used when transmitting SNMPv3 packets. Should a master failover occur, a different engine ID is transmitted. You can modify this behavior by manually assigning all stack members the same engine ID using the **snmp-server engineID local** command. However, should you decide to separate the stack and use the switches individually, you must remember to change or remove this configuration to prevent conflicts.

Example To set the SNMPv3 engine ID to 800000cf030000cd123456, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server engineID local
800000cf030000cd123456
```

To set a user defined SNMPv3 engine ID back to a system generated value, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server engineID local
```

Output The following example shows the engine ID values after configuration:

```
awplus(config)#snmp-server engineid local asdgdfh231234d
awplus(config)#exit
awplus#show snmp-server

SNMP Server ......................... Enabled
IP Protocol ......................... IPv4
SNMPv3 Engine ID (configured name) ... asdgdfh231234d
SNMPv3 Engine ID (actual) ............ 0x80001f888029af52e149198483

awplus(config)#no snmp-server engineid local
awplus(config)#exit
awplus#show snmp-server

SNMP Server ......................... Enabled
IP Protocol ......................... IPv4
SNMPv3 Engine ID (configured name) ... Not set
SNMPv3 Engine ID (actual) ............ 0x80001f888029af52e149198483
```

Validation Commands **show snmp-server**

Related Commands **snmp-server engineID local reset**
snmp-server group

snmp-server engineID local reset

Use this command to force the switch to generate a new pseudo-random SNMPv3 engine ID by resetting the SNMPv3 engine. If the current engine ID is user defined, use the **snmp-server engineID local** command to set SNMPv3 engine ID to a system generated value.

Syntax `snmp-server engineID local reset`

Mode Global Configuration

Example To force the SNMPv3 engine ID to be reset to a system generated value, use the commands:

```
awplus# configure terminal
awplus(config)# snmp-server engineID local reset
```

Validation Commands **show snmp-server**

Related Commands **snmp-server engineID local**

snmp-server group

This command is used with SNMP version 3 only, and adds an SNMP group, optionally setting the security level and view access modes for the group. The security and access views defined for the group represent the minimum required of its users in order to gain access.

The **no** variant of this command deletes an SNMP group, and is used with SNMPv3 only. The group with the specified authentication/encryption parameters must already exist.

Syntax

```
snmp-server group <groupname> {auth|noauth|priv} [read <readname>|
    write <writename>|notify <notifyname>]

no snmp-server group <groupname> {auth|noauth|priv}
```

Parameter	Description
<groupname>	Group name. The group name is a string up to 20 characters long and is case sensitive.
`auth`	Authentication.
`noauth`	No authentication and no encryption.
`priv`	Authentication and encryption.
`read`	Configure read view.
<readname>	Read view name.
`write`	Configure write view.
<writename>	Write view name. The view name is a string up to 20 characters long and is case sensitive.
`notify`	Configure notify view.
<notifyname>	Notify view name. The view name is a string up to 20 characters long and is case sensitive.

Mode Global Configuration

Examples To add SNMP group, for ordinary users, user the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server group usergroup noauth read
useraccess write useraccess
```

To delete SNMP group `usergroup`, use the following commands

```
awplus# configure terminal
awplus(config)# no snmp-server group usergroup noauth
```

Related Commands

snmp-server
show snmp-server
show snmp-server group
show snmp-server user

snmp-server host

This command specifies an SNMP trap host destination to which Trap or Inform messages generated by the device are sent.

For SNMP version 1 and 2c you must specify the community name parameter. For SNMP version 3, specify the authentication/encryption parameters and the user name. If the version is not specified, the default is SNMP version 1. Inform messages can be sent instead of traps for SNMP version 2c and 3.

Use the **no** variant of this command to remove an SNMP trap host. The trap host must already exist.

The trap host is uniquely identified by:

- host IP address (IPv4 or IPv6),
- inform or trap messages,
- community name (SNMPv1 or SNMP v2c) or the authentication/encryption parameters and user name (SNMP v3).

Syntax

```
snmp-server host {<ipv4-address>|<ipv6-address>} [traps] [version 1]
   <community-name>]

snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
   version 2c <community-name>

snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
   version 3 {auth|noauth|priv} <user-name>

no snmp-server host {<ipv4-address>|<ipv6-address>} [traps]
   [version 1] <community-name>

no snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
   version 2c <community-name>

no snmp-server host {<ipv4-address>|<ipv6-address>} [informs|traps]
   version 3 {auth|noauth|priv} <user-name>
```

Parameter	Description
<ipv4-address>	IPv4 trap host address in the format A.B.C.D, for example, 192.0.2.2.
<ipv6-address>	IPv6 trap host address in the format x:x::x:x for example, 2001:db8::8a2e:7334.
informs	Send Inform messages to this host.
traps	Send Trap messages to this host (default).
version	SNMP version to use for notification messages. Default: version 1.
1	Use SNMPv1(default).
2c	Use SNMPv2c.
3	Use SNMPv3.
auth	Authentication.
noauth	No authentication.

Parameter(cont.)	Description(cont.)
`priv`	Encryption.
<community-name>	The SNMPv1 or SNMPv2c community name.
<user-name>	SNMPv3 user name.

Mode Global Configuration

Examples To configure the device to send generated traps to the IPv4 host destination `192.0.2.5` with the SNMPv2c community name *public*, use the following command:

```
awplus# configure terminal
awplus(config)# snmp-server host 192.0.2.5 version 2c public
```

To configure the device to send generated traps to the IPv6 host destination `2001:db8::8a2e:7334` with the SNMPv2c community name *private*, use the following command:

```
awplus# configure terminal
awplus(config)# snmp-server host 2001:db8::8a2e:7334 version 2c
private
```

To remove a configured trap host of 192.0.2.5 with the SNMPv2c community name *public*, use the following command:

```
awplus# configure terminal
awplus(config)# no snmp-server host 192.0.2.5 version 2c public
```

Related Commands **snmp trap link-status**
snmp-server enable trap
snmp-server view

snmp-server location

This command sets the location of the system. The location is:

- displayed in the output of the **show system** command
- stored in the MIB object sysLocation

The **no** variant of this command removes the configured location from the system.

Syntax

```
snmp-server location <location-name>
no snmp-server location
```

Parameter	Description
<location-name>	The location of the system, from 0 to 255 characters long. Valid characters are any printable character and spaces.

Mode Global Configuration

Example To set the location to "server room 523", use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server location server room 523
```

Related Commands

show snmp-server
show system
snmp-server contact

snmp-server source-interface

Use this command to specify the interface that SNMP traps or informs originate from. You cannot specify an interface that does not already have an IP address assigned to the interface.

Use the **no** variant of this command to reset to the default source interface that SNMP traps or informs originate from (the Egress interface as sent from by default).

Syntax

```
snmp-server source-interface {traps|informs} <interface-name>
no snmp-server source-interface {traps|informs}
```

Parameter	Description
`traps`	SNMP traps.
`informs`	SNMP informs.
<interface-name>	Interface name (with an IP address already assigned).

Default By default the source interface is the Egress interface where traps or informs were sent from.

Mode Global Configuration

Usage An SNMP trap or inform sent from an SNMP server has the notification IP address of the interface where it was sent from. Use this command to monitor notifications from an interface.

Example To set the interface that SNMP informs originate from to port 1.0.2 for inform packets, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server source-interface informs port1.0.2
```

To reset the interface to the default source interface (the Egress interface) that SNMP traps originate from for trap packets, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server source-interface traps
```

Validation Commands **show running-config**

snmp-server startup-trap-delay

Use this command to set the time in seconds after following completion of the switch startup sequence before the switch sends any SNMP traps (or SNMP notifications).

Use the **no** variant of this command to restore the default startup delay of 30 seconds.

Syntax

```
snmp-server startup-trap-delay <delay-time>

no snmp-server startup-trap-delay
```

Parameter	Description
<delay-time>	Specify an SNMP trap delay time in seconds in the range of **30 to 600** seconds.

Default The SNMP server trap delay time is 30 seconds. The **no** variant restores the default.

Mode Global Configuration

Example To delay the switch sending SNMP traps until 60 seconds after switch startup, use the following commands:

```
awplus# configure terminal
awplus(config)# snmp-server startup-trap-delay 60
```

To restore the sending of SNMP traps to the default of 30 seconds after switch startup, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server startup-trap-delay
```

Validation Commands **show snmp-server**

snmp-server user

Use this command to create or move users as members of specified groups. This command is used with SNMPv3 only.

The **no** variant of this command removes an SNMPv3 user. The specified user must already exist.

Syntax

```
snmp-server user <username> <groupname> [encrypted] [auth {md5|
    sha} <auth-password>] [priv {des|aes} <privacy-password>]

no snmp-server user <username>
```

Parameter	Description
<username>	User name. The user name is a string up to 20 characters long and is case sensitive.
<groupname>	Group name. The group name is a string up to 20 characters long and is case sensitive.
`encrypted`	Use the encrypted parameter when you want to enter encrypted passwords.
`auth`	Authentication protocol.
`md5`	MD5 Message Digest Algorithms.
`sha`	SHA Secure Hash Algorithm.
<auth-password>	Authentication password. The password is a string of 8 to 20 characters long and is case sensitive.
`priv`	Privacy protocol.
`des`	DES Data Encryption Standard.
`aes`	AES Advanced Encryption Standards.
<privacy-password>	Privacy password. The password is a string of 8 to 20 characters long and is case sensitive.

Mode Global Configuration

Usage Additionally this command provides the option of selecting an authentication protocol and (where appropriate) an associated password. Similarly, options are offered for selecting a privacy protocol and password.

- Note that each SNMP user must be configured on both the manager and agent entities. Where passwords are used, these passwords must be the same for both entities.
- Use the **encrypted** parameter when you want to enter already encrypted passwords in encrypted form as displayed in the running and startup configs stored on the switch. For example, you may need to move a user from one group to another group and keep the same passwords for the user instead of removing the user to apply new passwords.
- User passwords are entered using plaintext without the **encrypted** parameter and are encrypted according to the authentication and privacy protocols selected.

- User passwords are viewed as encrypted passwords in running and startup configs shown from **show running-config** and **show startup-config** commands respectively. Copy and paste encrypted passwords from running-configs or startup-configs to avoid entry errors.

Examples To add SNMP user `authuser` as a member of group `usergroup`, with authentication protocol `md5`, authentication password `Authpass`, privacy protocol `des` and privacy password `Privpass`, use the following commands

```
awplus# configure terminal
awplus(config)# snmp-server user authuser usergroup auth md5
Authpass priv des Privpass
```

Validate the user is assigned to the group using the following command:

```
awplus#show snmp-server user
Name                Group name          Auth        Privacy
-------             -------------       -------     ----------
authuser            usergroup           md5         des
```

To enter existing SNMP user `authuser` with existing passwords as a member of group `newusergroup` with authentication protocol `md5` plus the encrypted authentication password `0x1c74b9c22118291b0ce0cd883f8dab6b74`, privacy protocol `des` plus the encrypted privacy password `0x0e0133db5453ebd03822b004eeacb6608f`, use the following commands

```
awplus# configure terminal
awplus(config)# snmp-server user authuser newusergroup
encrypted auth md5
0x1c74b9c22118291b0ce0cd883f8dab6b74 priv des
0x0e0133db5453ebd03822b004eeacb6608f
```

Note Copy and paste the encrypted passwords from the **running-config** or the **startup-config** displayed, using the **show running-config** and **show startup-config** commands respectively, into the command line to avoid key stroke errors issuing this command.

Validate the user has been moved from the first group using the following command:

```
awplus#show snmp-server user
Name                Group name          Auth        Privacy
-------             -------------       -------     ----------
authuser            newusergroup        md5         des
```

To delete SNMP user `authuser`, use the following commands:

```
awplus# configure terminal
awplus(config)# no snmp-server user authuser
```

Related Commands **show snmp-server user**
snmp-server view

snmp-server view

Use this command to create an SNMP view that specifies a sub-tree of the MIB. Further sub-trees can then be added by specifying a new OID to an existing view. Views can be used in SNMP communities or groups to control the remote manager's access.

Note The object identifier must be specified in a sequence of integers separated by decimal points.

The **no** variant of this command removes the specified view on the device. The view must already exist.

Syntax

```
snmp-server view <view-name> <mib-name> {included|excluded}
no snmp-server view <view-name>
```

Parameter	Description
<view-name>	SNMP server view name. The view name is a string up to 20 characters long and is case sensitive.
<mib-name>	Object identifier of the MIB.
`included`	Include this OID in the view.
`excluded`	Exclude this OID in the view.

Mode Global Configuration

Examples The following command creates a view called "loc" that includes the system location MIB sub-tree.

```
awplus(config)# snmp-server view loc 1.3.6.1.2.1.1.6.0 included
```

To remove the view "loc" use the following command

```
awplus(config)# no snmp-server view loc
```

Related Commands **show snmp-server view**
snmp-server community

undebug snmp

This command applies the functionality of the no **debug snmp** command.

Chapter 95: SNMP MIBs

Introduction 95.2
About MIBs 95.2
About SNMP 95.2
Obtaining MIBs 95.2
Loading MIBs 95.3
Allied Telesis Enterprise MIB 95.5
AT-ALMMON-MIB 95.6
AT-ATMF-MIB 95.8
AT-BOARDS-MIB 95.13
AT-DHCPSN-MIB 95.17
AT-DNS-CLIENT-MIB 95.20
AT-ENVMONv2-MIB 95.21
AT-EPSRv2-MIB 95.29
AT-FILEv2-MIB 95.31
AT-IP-MIB 95.38
AT-LICENSE-MIB 95.40
AT-LOG-MIB 95.43
AT-LOOPPROTECT-MIB 95.45
AT-MIBVERSION-MIB 95.47
AT-NTP-MIB 95.48
AT-PRODUCTS-MIB 95.51
AT-RESOURCE-MIB 95.54
AT-SETUP-MIB 95.55
AT-SMI-MIB 95.63
AT-SYSINFO-MIB 95.65
AT-TRIGGER-MIB 95.69
AT-USER-MIB 95.71
AT-VCSTACK-MIB 95.73
AT-VLANINFO-MIB 95.78
Other Enterprise MIBs 95.80
sFlow-MIB 95.80
Public MIBs 95.81

Introduction

This chapter describes the Management Information Bases (MIBs) and managed objects supported by the AlliedWare Plus™ Operating System. The following topics are covered:

- **"Allied Telesis Enterprise MIB" on page 95.5** describes the objects implemented in the Allied Telesis Enterprise MIB
- **"Public MIBs" on page 95.81** describes the public MIBs supported by the AlliedWare Plus™ Operating System, and any variations from the standard implementation.

About MIBs

A MIB is a collection of managed objects organized into a tree-like hierarchy of nodes in which the managed objects form the leaves. Within the tree, each node is identified by a non-negative integer identifier that is unique among the node's siblings. The address, or object identifier, of any node within the tree is expressed as a series of dot-delimited node identifiers that trace the path from the root of the tree to the node. For example, the object identifier for the sysDescr object is 1.3.6.1.2.1.1.1.

For more information about MIBs and the structure of management information, see **Chapter 93, SNMP Introduction**.

Note This chapter does not indicate which MIB objects are not-accessible (and therefore cannot be queried directly). Please consult the MIB files for that information.

About SNMP

A network management station (NMS) uses a protocol known as Simple Network Management Protocol (SNMP) to query or change the values of objects in the MIB of managed devices.

A managed device uses SNMP to respond to queries from an NMS, and to send unsolicited alerts (traps) to an NMS in response to events.

For more information about the Simple Network Management Protocol (SNMP), see **Chapter 93, SNMP Introduction**.

For information about configuring SNMP, see **Chapter 94, SNMP Commands**.

Obtaining MIBs

You can download MIBs from the following locations:

Download this MIB...	From this location...
Allied Telesis Enterprise MIB	The MIB files are available with the software files from the Support area at **www.alliedtelesis.com/support/software**
Public MIBs defined in RFCs	**www.rfc-editor.org/rfc.html**
IANAifType-MIB	**www.iana.org/assignments/ianaiftype-mib**

Loading MIBs

Individual MIBs define a portion of the total MIB for a device. For example, the MAU-MIB defines objects for managing IEEE 802.3 medium attachment units (MAUs), and forms a sub-tree under mib-2 with the object identifier snmpDot3MauMgt (1.3.6.1.2.1.26).

All the objects within a MIB are assigned object identifiers relative to a parent object. Most MIBs import the object identifier of the parent object, along with other object identifiers, textual conventions, macros and syntax types from the MIBs where they are defined. This creates dependencies between MIBs.

Some network management stations and MIB compilers will generate errors if you load a MIB that depends on another MIB that has not already been loaded. To avoid these errors, we recommend that you load MIBs in the following order:

1. RFC 1212
   RFC 1239
   RFC 2257
   RFC 3410

2. RFC1155-SMI (RFC 1155)
   SNMPv2-SMI (RFC 2578)
   SNMPv2-PDU (RFC 3416)

3. RFC1213-MIB (RFC 1213)
   RFC 1215
   SNMPv2-TC (RFC 2579)
   SNMPv2-CONF (RFC 2580)

4. IP-MIB (RFC 2011)
   TCP-MIB (RFC 2012)
   UDP-MIB (RFC 2013)
   IP-FORWARD-MIB (RFC 2096)
   SNMP-MPD-MIB (RFC 2572)
   RMON-MIB (RFC 2819)
   HCNUM-TC (RFC 2856)
   SNMP-FRAMEWORK-MIB (RFC 3411)
   SNMP-MPD-MIB (RFC 3412)
   SNMPv2-TM (RFC 3417)
   SNMPv2-MIB (RFC 3418)
   INET-ADDRESS-MIB (RFC 4001)
   IANAifType-MIB

5. IF-MIB (RFC 2863)
   SNMP-TARGET-MIB (RFC 3413)

6. SNMP-COMMUNITY-MIB (RFC 2576)
   EtherLike-MIB (RFC 3635)
   MAU-MIB (RFC 3636)
   BRIDGE-MIB (RFC 4188)
   DISMAN-PING-MIB (RFC 4560)

SNMP-NOTIFICATION-MIB (RFC 3413)
SNMP-PROXY-MIB (RFC 3413)

7. P-BRIDGE-MIB (RFC 2674)
   Q-BRIDGE-MIB (RFC 2674)
   RSTP-MIB (RFC 4318)
   LLDP-MIB
   LLDP-EXT-DOT1-MIB
   LLDP-EXT-DOT3-MIB
   LLDP-EXT-MED-MIB
   POE-MIB
   VRRPv3-MIB

8. AT-SMI-MIB

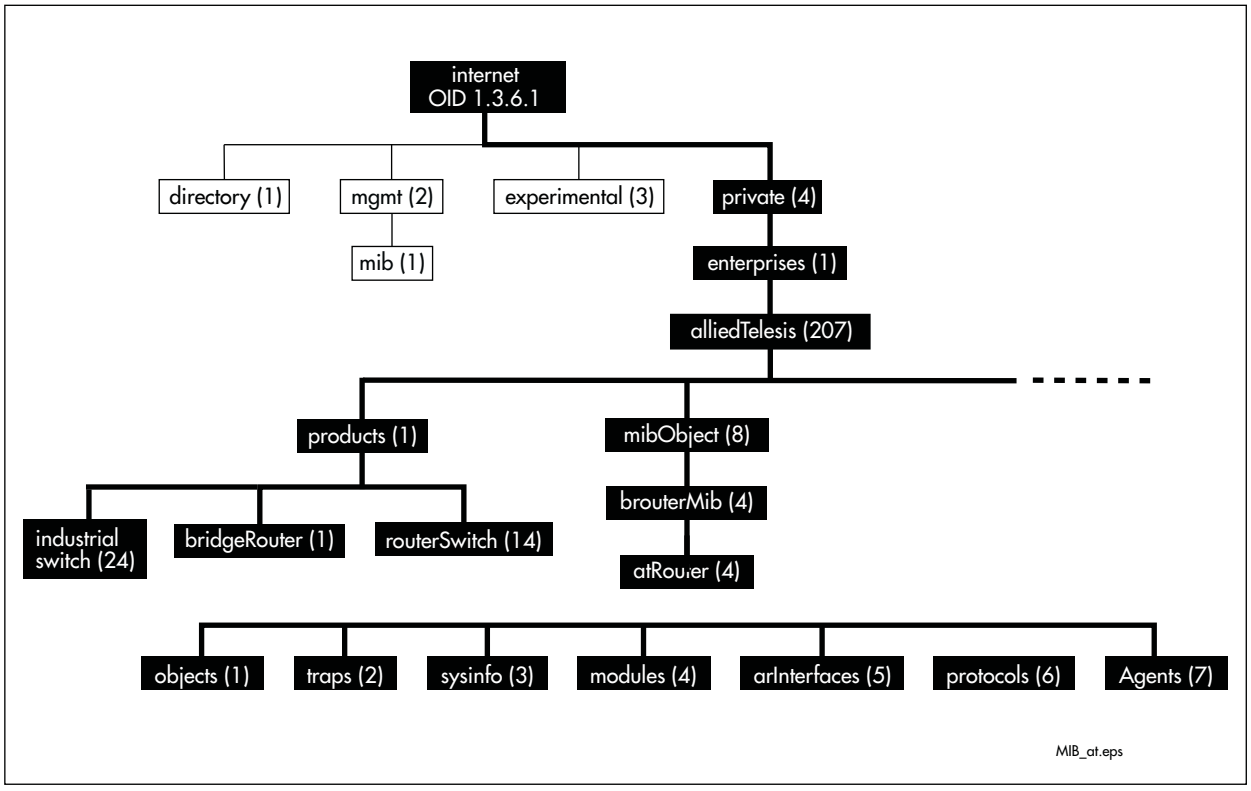
9. AT-BOARDS-MIB
   AT-PRODUCT-MIB
   AT-SETUP-MIB
   AT-SYSINFO-MIB
   AT-TRIGGER-MIB
   AT-VCSTACK-MIB
   AT-USER-MIB
   AT-RESOURCE-MIB
   AT-LICENSE-MIB
   AT-LOOPPROTECT-MIB
   AT-DNS-CLIENT--MIB
   AT-NTP-MIB
   AT-EPSRv2-MIB
   AT-FILEv2-MIB
   AT-LOG-MIB
   AT-IP-MIB
   AT-ENVMONv2-MIB
   AT-MIBVERSION-MIB
   AT-DHCPSN-MIB
   AT-ALMMON-MIB

Allied Telesis Enterprise MIB

The *Allied Telesis Enterprise MIB* defines a portion of the Management Information Base (MIB) for managing Allied Telesis products and features that are not supported by public MIBs. Objects defined in this MIB reside in the private(4) subtree and have the object identifier alliedTelesis ({ enterprises 207 }).

This document describes only those portions of the Allied Telesis Enterprise MIB supported by the AlliedWare Plus™ Operating System. **Figure 95-1** shows the structure of the Allied Telesis Enterprise MIB. Each component MIB is detailed in the following sections of this chapter.

Figure 95-1: The Allied Telesis Enterprise MIB sub-tree of the Internet-standard Management Information Base (MIB)



AT-ALMMON-MIB

AT-ALMMON-MIB defines objects for managing alarms **Table 95-1**. Objects in this group have the object identifier sysinfo ({ atRouter 3 }).

Table 95-1: .Objects defined by the atAlmMon MIB

Object / Object Identifier	Description
atAlmMon { atAlmMon 24 } (OID 1.3.6.1.4.1.207.8.4.4.3.26)	The AT Alarm Monitoring v2 MIB for managing and reporting device alarms.
atAlmMonActionTable { atAlmMon 1} (207.8.4.4.3.26.1)	Table of information defining alarm monitoring inputs and consequent actions (i.e. fault LED and relay outputs), indexed by: 1. atAlmMonActionStackMemberId 2. atAlmMonActionIndex
atAlmMonActionEntry { atAlmMonActionTable 1 } (207.8.4.4.3.26.1.1)	A description and configuration of what to do for a specific monitored alarm.
atAlmMonActionStackMemberId { atAlmMonActionEntry 1 } (207.8.4.4.3.26.1.1.1)	The index of the stack member of this alarm action. Read only.
atAlmMonActionIndex { atAlmMonActionEntry 2 } (207.8.4.4.3.26.1.1.2)	The numeric identifier of this alarm action. Read only.
atAlmMonAlarmType { alAlmMonActionEntry 3 } (207.8.4.4.3.26.1.1.3)	The type of alarm that this action monitors. Read only. Values can be: 1. alarmTypeInvalid (0) 2. externalPSU (1) 3. epsr (2) 4. contactInput (3) 5. portLinkDown (4) 6. loopDetect (5) 7. mainPse (6) 8. portPoeFailure (7) 9. temperature (8)
atAlmMonAlarmTypeSelection { atAlmMonActionEntry 4}	The 1-based index of the alarm of the particular type (as categorized by AlmMonAlarmType).
atAlmMonActionDescription { atAlmMonActionEntry 5}	The description of this alarm monitoring entry. Read only.
atAlmMonActionUseRelay1 { atAlmMonActionEntry 6 }	Indicates/controls whether or not this alarm monitor drives the first relay output. Read-write. Values can be: ■ Unused (1) ■ Used (2)
atAlmMonActionUseRelay2 { alAlmMonActionEntry 7 }	Indicates/controls whether or not this alarm monitor drives the second relay output. Read-write. Values can be: ■ Unused (1) ■ Used (2)

Object / Object Identifier		Description
	atAlmMonActionUseRelay3 { atAlmMonActionEntry 8 }	Indicates/controls whether or not this alarm monitor drives the third relay output. Read-write. Values can be: ■ Unused (1) ■ Used (2)
	atAlmMonActionUseFaultLed { alAlmMonActionEntry 9 }	Indicates/controls whether or not this alarm monitor drives the fault LED. Read-write. Values can be: ■ Unused (1) ■ Used (2)
	atAlmMonAbnormalState { atAlmMonActionEntry 10 }	Indicates/sets the abnormal (i.e., alarm active) state for a contact input. Only used for contactInput alarm monitors, ignored for all other types. Read-write. Values can be: ■ open (1) ■ closed (2)
	atAlmMonActionState { atAlmMonActionEntry 11 }	Indicates the current state of this alarm monitor. Read only. Values can be: ■ Inactive (1) ■ Active (2)

AT-ATMF-MIB

[Section written by PSN 2008 & edited in Jan 09]

The ATMF-MIB defines objects for managing ATMF. Objects and triggers are shown diagrammatically in **Figure 95-2 on page 95.8** and **Figure 95-3 on page 95.9**. Objects in this group have the object identifier ATMF ({ modules 603 }). These are shown listed in **Figure 95-2 on page 95.10**.

Figure 95-2: The upper levels of the AT-ATMF MIB sub-tree

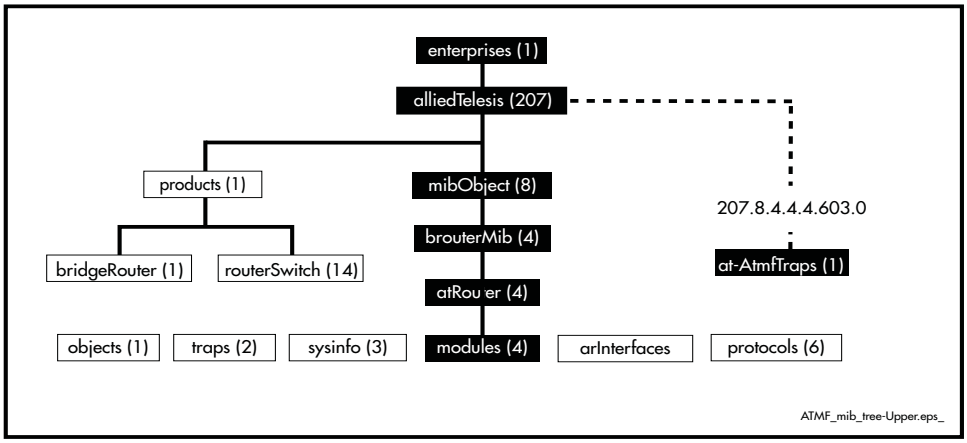


Figure 95-3: The Lower levels of the AT-ATMF MIB sub-tree

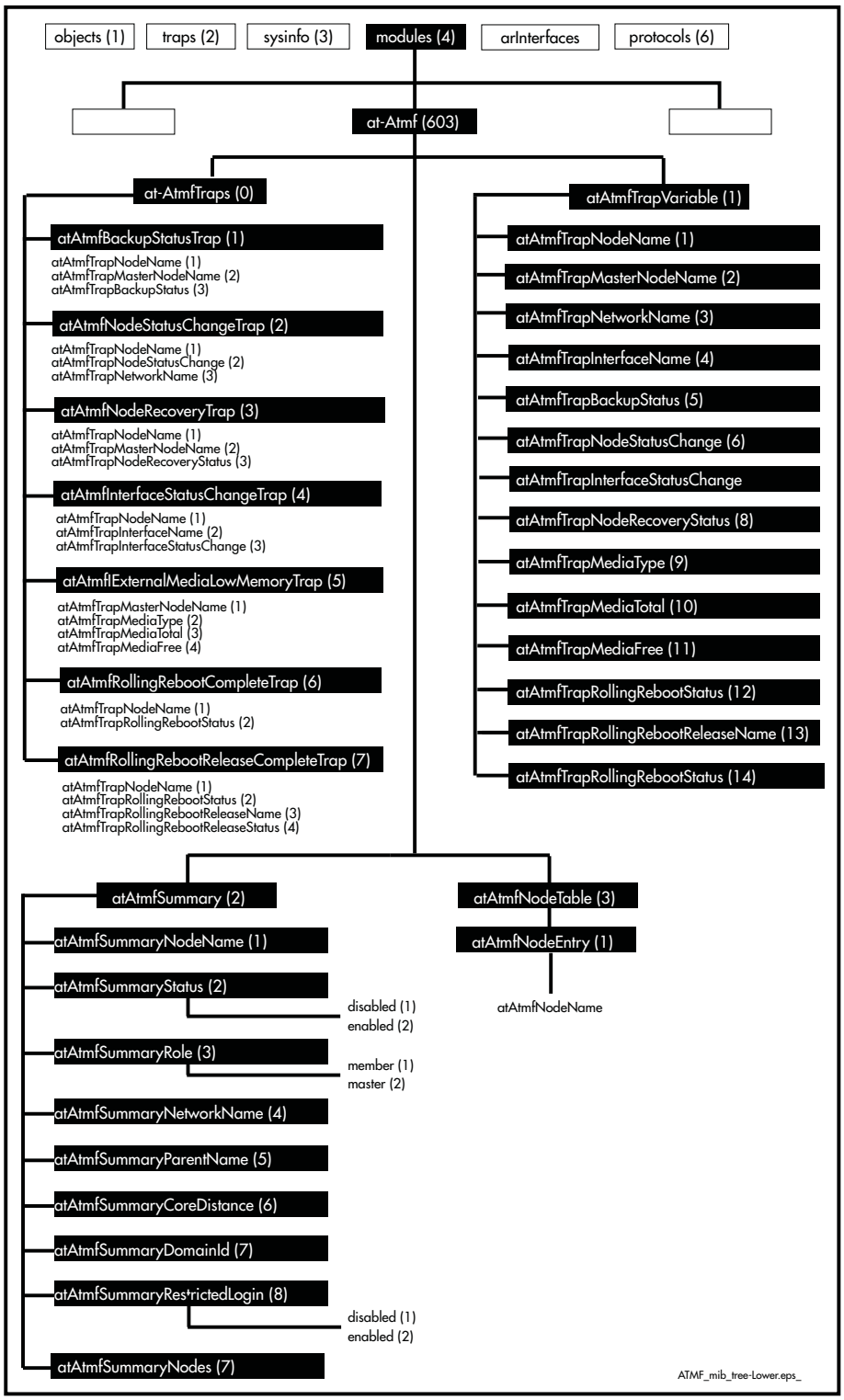


Table 95-2: AT-ATMF MIB Objects

Object	Object Identifier	Description
atmf	{ modules 603 } 1.3.6.1.4.1.207.8.4.4.4.603	Root of the Allied Telesis ATMF MIB under the private(4) node defined in RFC1155-SMI.
atAtmfTraps	{ atmf 0 } 207.8.4.4.4.603.0	Sub-tree of objects describing ATMF traps.
atAtmfBackup StatusTrap	atAtmfTraps 1	This trap is generated when an ATMF master attempts to backup a node's FLASH contents. It states whether the backup of an individual node, or all nodes, to a master node has passed or failed. Its objects are: 1. atAtmfTrapNodeName 2. atAtmfTrapMasterNodeName 3. atAtmfTrapBackupStatus
atAtmfNode StatusChange Trap	atAtmfTraps 2	This trap is generated when an ATMF node joins or leaves the ATMF network. It states whether a node has <left|joined> an ATMF network. Its objects are: 1. atAtmfTrapNodeName 2. atAtmfTrapNodeStatusChange 3. atAtmfTrapNetworkName
atAtmfNode RecoveryTrap	atAtmfTraps 3	This trap is generated when an attempt has been made to recover an ATMF node. It states whether an attempt to recover a node from the specified master has passed or failed. Its objects are: 1. atAtmfTrapNodeName 2. atAtmfTrapMasterNodeName 3. atAtmfTrapNodeRecoveryStatus
atAtmfInterface StatusChangeTrap	atAtmfTraps 4	This trap is generated when an ATMF interface status change occurs. It states that an interface on a node has changed status to either blocking or forwarding. Its objects are: 1. atAtmfTrapNodeName 2. atAtmfTrapInterfaceName 3. atAtmfTrapInterfaceStatusChange
atAtmfExternal MediaLowMemory Trap	atAtmfTraps 5	This trap is generated when the available external storage on the ATMF master node falls below a nominated threshold. It states that the external USB or SD card storage on a master node has fallen below the designated threshold and specifies the total available memory <xxx MB> and the total free memory <xxx MB>. Its objects are: 1. atAtmfTrapMasterNodeName 2. atAtmfTrapMediaType 3. atAtmfTrapMediaTotal 4. atAtmfTrapMediaFree
atAtmfRolling RebootComplete Trap	atAtmfTraps 6	This trap is generated when the ATMF rolling reboot process has finished on a particular ATMF node. Nominally, it states that the ATMF rolling reboot, executed against the specified node, has returned a reboot status of either failed or passed. Its objects are: 1. atAtmfTrapNodeName 2. atAtmfTrapRollingRebootStatus

Table 95-2: AT-ATMF MIB Objects(cont.)

Object	Object Identifier	Description
atAtmfRolling RebootRelease CompleteTrap	atAtmfTraps 7	This trap is generated when the ATMF rolling reboot process attempts to push a new software release to a specified ATMF node. Nominally, it states that the ATMF rolling reboot release process, executed from the specified node has returned a reboot status of either failed or passed, the name of the attempted release file and the release status of either "failed" or "passed". Its objects are: 1. atAtmfTrapNodeName 2. atAtmfTrapRollingRebootStatus 3. atAtmfTrapRollingRebootReleaseName 4. atAtmfTrapRollingRebootReleaseStatus
atAtmfTrapVariable	atmf 1	Sub-tree of objects describing ATMF traps.
atAtmfTrapNode Name	atAtmfTrapVariable 1	The ATMF trap node name.
atAtmfTrapMaster NodeName	atAtmfTrapVariable 2	The ATMF trap master node name.
atAtmfTrapNetwork Name	atAtmfTrapVariable 3	The ATMF trap network name.
atAtmfTrapInterface Name	atAtmfTrapVariable 4	The ATMF interface name, "Trap".
atAtmfTrapBackup Status	atAtmfTrapVariable 5	The status of the last trap backup attempt on either a specified ATMF node or all nodes in the ATMF network. Its objects are: 1. failed(1) 2. passed(2)
atAtmfTrapNode StatusChange	atAtmfTrapVariable 6	An ATMF trap node has changed its status in the ATMF network. Its objects are: 1. left(1) 2. joined(2)
AtmfTrapInterface StatusChange	atAtmfTrapVariable 7	An ATMFtrap interface has changed its status. Its objects are: 1. blocking(1) 2. forwarding(2)
atAtmfTrapNode RecoveryStatus	atAtmfTrapVariable 8	The status of the last recovery attempt. Its objects are: 1. failed(1) 2. passed(2)
atAtmfTrapMedia Type	atAtmfTrapVariable 9	The media type resident on the ATMF node - USB or SD.
atAtmfTrapMedia Total	atAtmfTrapVariable 10	The total memory available on the resident media, in MB.
atAtmfTrapMedia Free	atAtmfTrapVariable 11	The free memory available on the resident media, in MB. Each node has a maximum flash of 64MB.
atAtmfTrapRolling RebootStatus	atAtmfTrapVariable 12	The status of the last rolling reboot for a node. Its objects are: 1. failed(1) 2. passed(2)
atAtmfTrapRolling RebootRelease Name	atAtmfTrapVariable 13	The name of the last rolling reboot release.

Table 95-2: AT-ATMF MIB Objects(cont.)

Object	Object Identifier	Description
atAtmfTrapRollingRebootReleaseStatus	atAtmfTrapVariable 14	The release update status of the last rolling reboot for a node. Its objects are: 1. failed(1) 2. passed(2)
atAtmfSummary	atmf 2	
atAtmfSummary NodeName	atAtmfSummary 1	The name assigned to a particular node.
atAtmfSummary Status	atAtmfSummary 2	The Node's ATMF status.
atAtmfSummary Role	atAtmfSummary 3	The role configured for this ATMF device, either Master or Member.
atAtmfSummary NetworkName	atAtmfSummary 4	The ATMF network that a particular node belongs to.
atAtmfSummary ParentName	atAtmfSummary 5	The parent name of the node or 'none'.
atAtmfSummary CoreDistance	atAtmfSummary 6	The ATMF core distance for this node.
atAtmfSummary DomainId	atAtmfSummary 7	The ATMF domain Id for this node.
atAtmfSummary RestrictedLogin	atAtmfSummary 8	The login for this ATMF device is restricted to only those devices that are designated ATMF Masters. Its objects are: 1. disabled(1) 2. enabled(2)
atAtmfSummary Nodes	atAtmfSummary 9	The number of ATMF nodes known to this device.
atAtmfNodeTable	atmf 3	ATMF Node Entry.
atAtmfNodeName	atAtmfNodeTable 1	The name assigned to a particular node.

AT-BOARDS-MIB

AT-BOARDS-MIB defines object identifiers for components of Allied Telesis products—base CPU and expansion boards, interface types, and chip sets. Objects in this MIB have the object identifier objects ({ atRouter 1 }), and are organized into the following groups:

- Base CPU and expansion boards (**Table 95-3**). These object identifiers are for use with the hrDeviceID object in the Host Resources MIB (see **"Public MIBs" on page 95.81**).
- Interface types (**Table 95-4**).
- Chip sets (**Table 95-5**).

Table 95-3: Object identifiers for base CPU and expansion boards

Object	Object Identifier	Description
boards	{ objects 1 }	
pprx90024XT	{ boards 271 }	x900-24XT Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays
pprx90024XS	{ boards 272 }	x900-24XS Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays
pprAtXum10Gi	{ boards 273 }	XEM-1XP Expansion Module, 1 x 10Gbe XFP port
pprAtXum12SFPi	{ boards 274 }	XEM-12S Expansion Module, 12 x SFP Gigabit ports
pprAtXum12Ti	{ boards 275 }	XEM-12T Expansion Module, 12 x 10/100/100BASE-T copper ports (RJ-45 connectors)
pprAtXum12TiN	{ boards 280 }	XEM-12T-N Expansion Module, 12 x 10/100/100BASE-T copper ports (RJ-45 connectors), NEBS compliant
pprx90024XTN	{ boards 281 }	x900-24XT Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays, NEBS compliant
pprSwitchBladex908	{ boards 282 }	Switchblade x908 8 Slot Layer 3 Switch Chassis
pprx90012XTS	{ boards 288 }	AT-x900-12XT/S Advanced Gigabit Layer 3+ Expandable Switch, 12 x combo ports (10/100/1000BASE-T copper or SFP), 1 x 30Gbps expansion bay
pprAt9524TS	{ boards 290 }	x600-24Ts/XP, 24 x 1000BASE-T ports (RJ45 connectors), 4 x SFP (combo) ports
pprAt9524TSXP	{ boards 291 }	x600-24Ts/XP, 24 x 1000BASE-T ports (RJ45 connectors), 4 x SFP (combo) ports, 2 x XFP ports
pprAt9548TS	{ boards 294 }	x600-44Ts, 44 x 1000BASE-T ports, 4 x SFP ports
pprAt9548TSXP	{ boards 295 }	x600-44Ts/XP, 44 x 1000BASE-T ports, 4 x SFP ports, 2 x XFP ports
pprXem2XP	{ boards 306 }	XEM-2XP Expansion Module, 2 x 10Gbe XFP port
pprATStackXG	{ boards 307 }	x600 Expansion Module, Stacking
pprATEMXP	{ boards 308 }	x600 Expansion Module, 2 x 10G XFP ports
pprATLBM	{ boards 309 }	x600 Expansion Module, loopback
pprAtSBx8112	{ boards 316 }	AT-SBx8112, SwitchBlade x8112 chassis
pprAtSBx81CFC400	{ boards 317 }	AT-SBx81CFC, Control Fabric Card for SwitchBlade x8112
pprAtSBx81GP24	{ boards 318 }	AT-SBx81GP24, 24 x 1G PoE line card
pprAtSBxPWRSYSAC	{ boards 320 }	AT-SBxPWR SYS/AC, system power supply unit for the SwitchBlade x8112 (AC input)

Table 95-3: Object identifiers for base CPU and expansion boards(cont.)

Object	Object Identifier	Description
pprAtSBxPWRPOEAC	{ boards 321 }	AT-SBxPWR POE/AC, PoE power supply unit for the SwitchBlade x8112 (AC input)
pprAtSBxFAN12	{ boards 322 }	AT-SBxFAN12, fan tray for the SwitchBlade x8112
pprAtPWR05DC	{ boards 323 }	AT-PWR05, DC power supply unit for SwitchBlade x908
pprXem2XT	{ boards 325 }	XEM-2XT Expansion Module, 2 x 10Gbe copper XEM port
pprx60024TSPOE	{ boards 326 }	x600-24Ts-POE, 24 x 1000BASE-T PoE ports (RJ45 connectors), 4 x SFP (combo) ports
pprx60024TSPOEPLUS	{ boards 327 }	x600-24Ts-POE+, 24 x 1000BASE-T PoE+ ports (RJ45 connectors), 4 x SFP (combo) ports
pprx61048TsXPOEPlus	{ boards 331 }	x610-48Ts/X-POE+, 48 x 1000BASE-T PoE+ ports (RJ45 connectors), 2 x SFP (combo) ports, 2 x SFP+ ports
pprx61048TsPOEPlus	{ boards 332 }	x610-48Ts-POE+, 48 x 1000BASE-T PoE+ ports (RJ45 connectors), 4 x SFP (combo) ports
pprx61024TsXPOEPlus	{ boards 333 }	x610-24Ts/X-POE+, 24 x 1000BASE-T PoE+ ports (RJ45 connectors), 4 x SFP (combo) ports, 2 x SFP+ ports
pprx61024TsPOEPlus	{ boards 334 }	x610-24Ts-POE+, 24 x 1000BASE-T PoE+ ports (RJ45 connectors), 4 x SFP (combo) ports
pprPWR800	{ boards 336 }	AT-PWR800, 800W power supply unit
pprPWR1200	{ boards 337 }	AT-PWR1200, 1200W power supply unit
pprPWR250	{ boards 338 }	AT-PWR250, 250W power supply unit
pprx61048TsX	{ boards 339 }	x610-48Ts/X, 48 x 1000BASE-T ports (RJ45 connectors), 2 x SFP (combo) ports, 2 x SFP+ ports
pprx61048Ts	{ boards 340 }	x610-48Ts, 48 x 1000BASE-T ports (RJ45 connectors), 4 x SFP (combo) ports
pprx61024TsX	{ boards 341 }	x610-24Ts/X, 24 x 1000BASE-T ports (RJ45 connectors), 4 x SFP (combo) ports, 2 x SFP+ ports
pprx61024Ts	{ boards 342 }	x610-24Ts, 24 x 1000BASE-T ports (RJ45 connectors), 4 x SFP (combo) ports
pprPWR250DC	{ boards 351 }	AT-PWR250DC, 250W DC power supply unit
pprAtSBx81GT24	{ boards 352 }	AT-SBx81GT24, 24 x 1G copper line card
pprAtSBx81GS24a	{ boards 353 }	AT-SBx81GS24a, 24 x 1G SFP line card
pprAtSBx81XS6	{ boards 354 }	AT-SBx81XS6, 6 x 10G SFP+ line card
pprx2109GT	{ boards 367 }	AT-x210-9GT, 8xGigbit, 1xSFP/T
pprx21016GT	{ boards 368 }	AT-x210-16GT, 14xGigbit, 2xcombo SFP/T
pprx21024GT	{ boards 369 }	AT-x210-24GT, 20xGigbit, 4xcombo SFP/T
pprx51028GTX	{ boards 370 }	AT-x510-28GTX board with 24 10/100/1000 Base-T ports and four 10Gb/s SFP+ ports.
pprx51028GPX	{ boards 371 }	AT-x510-28GPX board with 24 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports and PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
pprx51028GSX	{ boards 372 }	AT-x510-28GSX with 24 100/1000 SFP ports and four 10 Gb/s SFP+ ports.
pprx51052GTX	{ boards 373 }	AT-x510-52GTX board with 48 10/100/1000 Base-T ports and four 10 Gb/s SFP+ ports.
pprx51052GPX	{ boards 374 }	AT-x510-52GPX board with 48 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports and PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.

Table 95-3: Object identifiers for base CPU and expansion boards(cont.)

Object	Object Identifier	Description
pprAtSBx8106	{ boards 375 }	AT-SBx8106, SwitchBlade x8106 chassis
pprAtSBxFAN06	{ boards 376 }	AT-SBxFAN06, fan tray for the SwitchBlade x8106
pprAtSBx81CFC960	{ boards 377 }	AT-SBx81CFC960 Control Fabric Card for SwitchBlade x8100 Series chassis, four 10GbE SFP+ ports.
pprAtSBx81GT40	{ boards 381 }	AT-SBx81GT40 RJ point five line card.
pprPWR100R	{ boards 384 }	AT-PWR100R, 100W power supply unit
pprPWR250DCR	{ boards 385 }	AT-PWR250R-DC, 250W DC power supply unit
pprx510DP52GTX	{ boards 386 }	AT-x510DP-52GTX board with 48 10/100/1000 Base-T ports and four 10 Gb/s SFP+ ports.
pprxIX528GPX	{ boards 387 }	AT-IX5-28GPX board with 24 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports and PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
pprx31026FT	{ boards 393 }	AT-x310-26FT board with 24 10/100 Base-T ports, two 10/100/1000 Base-T ports, and four 10 Gb/s SFP+ ports.
pprx31050FT	{ boards 394 }	AT-x310-50FT board with 48 10/100 Base-T ports, two 10/100/1000 Base-T ports, and four 10 Gb/s SFP+ ports.
pprx31026FP	{ boards 395 }	AT-x310-26FP board with 24 10/100 Base-T ports, two 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports, plus PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
pprx31050FP	{ boards 396 }	AT-x310-50FP board with 48 10/100 Base-T ports, two 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports, plus PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
pprIE2006GT	{ boards 410 }	IE200-6GT L2+ managed industrial Switch with 4 x 10/100/1000T LAN ports and 2 x SFP uplinks (100/1000X). Note that this is a single board device.
pprIE2006GP	{ boards 411 }	IE200-6GP L2+ managed industrial Switch with 4 x 10/100/1000T LAN ports (with 802.3at PoE+) and 2 x SFP uplinks (100/1000X). Note that this is a single board device.
pprIE2006GPW	{ boards 412 }	IE200-6GPW L2+ managed industrial Switch with 4 x 10/100/1000T LAN ports (with 802.3at PoE+) and 2 x SFP uplinks (100/1000X) and 802.11bgn wireless. Note that this is a single board device.

Table 95-4: Object identifiers for interface types

Object	Object Identifier	Description
iftypes	{ objects 3 }	
ifaceEth	{ iftypes 1 }	Ethernet
ifaceSyn	{ iftypes 2 }	Synchronous
ifaceAsyn	{ iftypes 3 }	Asynchronous
ifaceBri	{ iftypes 4 }	BRI ISDN
ifacePri	{ iftypes 5 }	PRI ISDN
ifacePots	{ iftypes 6 }	POTS (voice)
ifaceGBIC	{ iftypes 7 }	GBIC (Gigabit Interface Converter)
chipMips_4kcCpu	{ chips 6 }	Dual MIPS CPU

Table 95-5: Object identifiers for chip sets

Object	Object Identifier	Description
chips	{ objects 4 }	
chip68020Cpu	{ chips 1 }	MC68020 CPU
chip68340Cpu	{ chips 2 }	MC68340 CPU
chip68302Cpu	{ chips 3 }	MC68302 CPU
chip68360Cpu	{ chips 4 }	MC68360 CPU
chip860TCpu	{ chips 5 }	MPC860T CPU
chipMips4kcCpu	{ chips 6 }	Dual MIPS CPU
chipRtc1	{ chips 21 }	Real Time Clock v1
chipRtc2	{ chips 22 }	Real Time Clock v2
chipRtc3	{ chips 23 }	Real Time Clock v3
chipRtc4	{ chips 24 }	Real Time Clock v4
chipRam1mb	{ chips 31 }	1 MB RAM
chipRam2mb	{ chips 32 }	2 MB RAM
chipRam3mb	{ chips 33 }	3 MB RAM
chipRam4mb	{ chips 34 }	4 MB RAM
chipRam6mb	{ chips 36 }	6 MB RAM
chipRam8mb	{ chips 38 }	8 MB RAM
chipRam12mb	{ chips 42 }	12 MB RAM
chipRam16mb	{ chips 46 }	16 MB RAM
chipRam20mb	{ chips 50 }	20 MB RAM
chipRam32mb	{ chips 62 }	32 MB RAM
chipFlash1mb	{ chips 71 }	1 MB FLASH memory
chipFlash2mb	{ chips 72 }	2 MB FLASH memory
chipFlash3mb	{ chips 73 }	3 MB FLASH memory
chipFlash4mb	{ chips 74 }	4 MB FLASH memory
chipFlash6mb	{ chips 76 }	6 MB FLASH memory
chipFlash8mb	{ chips 78 }	8 MB FLASH memory
chipPem	{ chips 120 }	Processor Enhancement Module

AT-DHCPSN-MIB

This MIB contains objects for displaying and managing DHCP snooping and ARP security information on the switch. (**Table 95-6**). The objects reside in the module atDhcpsn { modules 537 }, organized in the following groups:

- The DHCP Snooping Events group (atDhcpsnEvents) contains notifications (traps)
- The DHCP Snooping table (atDhcpsnVariablesTable) contains DHCP snooping information
- The ARP Security table (atArpsecVariablesTable) contains ARP security information

Table 95-6: Objects defined in AT-DHCPSN-MIB

Object	Object Identifier	Description
atDhcpsn	{ modules 537 }	This MIB file contains definitions of managed objects for DHCP Snooping in AlliedWare Plus™.
atDhcpsnEvents	{ atDhcpsn 1 }	DHCP Snooping notifications (traps)
atDhcpsnTrap	{ atDhcpsnEvents 1 }	DHCP Snooping violation notification.
atArpsecTrap	{ atDhcpsnEvents 2 }	DHCP Snooping ARP Security violation notification.
atDhcpsnVariablesTable	{ atDhcpsn 1 }	The DHCP Snooping table. This table contains rows of DHCP Snooping information.
atDhcpsnVariablesEntry	{ atDhcpsnVariablesTable 1 }	A set of parameters that describe the DHCP Snooping features.
atDhcpsnIfIndex	{ atDhcpsnVariablesEntry 1 }	Ifindex of the port that the packet was received on.
atDhcpsnVid	{ atDhcpsnVariablesEntry 2 }	VLAN ID of the port that the packet was received on.
atDhcpsnSmac	{ atDhcpsnVariablesEntry 3 }	Source MAC address of the packet that caused the trap.
atDhcpsnOpcode	{ atDhcpsnVariablesEntry 4 }	Opcode value of the BOOTP packet that caused the trap. Only bootpRequest(1) or bootpReply(2) is valid.
atDhcpsnCiaddr	{ atDhcpsnVariablesEntry 5 }	Ciaddr value of the BOOTP packet that caused the trap.
atDhcpsnYiaddr	{ atDhcpsnVariablesEntry 6 }	Yiaddr value of the BOOTP packet that caused the trap.
atDhcpsnGiaddr	{ atDhcpsnVariablesEntry 7 }	Giaddr value of the BOOTP packet that caused the trap.
atDhcpsnSiaddr	{ atDhcpsnVariablesEntry 8 }	Siaddr value of the BOOTP packet that caused the trap.
atDhcpsnChaddr	{ atDhcpsnVariablesEntry 9 }	Chaddr value of the BOOTP packet that caused the trap.

Table 95-6: Objects defined in AT-DHCPSN-MIB(cont.)

Object	Object Identifier	Description
atDhcpsnVioType	{ atDhcpsnVariablesEntry 10 }	The reason that the trap was generated. ■ invalidBootp(1) indicates that the received BOOTP packet was invalid. For example, it is neither BootpRequest nor BootpReply. ■ invalidDhcpAck(2) indicates that the received DHCP ACK was invalid. ■ invalidDhcpRelDec(3) indicates the DHCP Release or Decline was invalid. ■ invalidIp(4) indicates that the received IP packet was invalid. ■ maxBindExceeded(5) indicates that if the entry was added, the maximum bindings configured for the port would be exceeded. ■ opt82InsertErr(6) indicates that the insertion of Option 82 failed. ■ opt82RxInvalid(7) indicates that the received Option 82 information was invalid. ■ opt82RxUntrusted(8) indicates that Option 82 information was received on an untrusted port. ■ opt82TxUntrusted(9) indicates that Option 82 would have been transmitted out an untrusted port. ■ replyRxUntrusted(10) indicates that a BOOTP Reply was received on an untrusted port. ■ srcMacChaddrMismatch(11) indicates that the source MAC address of the packet did not match the BOOTP CHADDR of the packet. ■ staticEntryExisted(12) indicates that the static entry to be added already exists. ■ dbAddErr(13) indicates that adding an entry to the database failed.
atArpsecVariablesTable	{ atDhcpsn 2 }	The ARP Security table. This table contains rows of DHCP Snooping ARP Security information.
atArpsecVariablesEntry	{ atArpsecVariablesTable 1 }	A set of parameters that describe the DHCP Snooping ARP Security features.
atArpsecIfIndex	{ atArpsecVariablesEntry 1 }	Ifindex of the port that the ARP packet was received on.
atArpsecClientIP	{ atArpsecVariablesEntry 2 }	Source IP address of the ARP packet.
atArpsecSrcMac	{ atArpsecVariablesEntry 3 }	Source MAC address of the ARP packet.
atArpsecVid	{ atArpsecVariablesEntry 4 }	VLAN ID of the port that the ARP packet was received on.

Table 95-6: Objects defined in AT-DHCPSN-MIB(cont.)

Object	Object Identifier	Description
atArpsecVioType	{ atArpsecVariablesEntry 5 }	The reason that the trap was generated. ■ srcIpNotFound(1) indicates that the Sender IP address of the ARP packet was not found in the DHCP Snooping database. ■ badVLAN(2) indicates that the VLAN of the DHCP Snooping binding entry associated with the Sender IP address of the ARP packet does not match the VLAN that the ARP packet was received on. ■ badPort(3) indicates that the port of the DHCP Snooping binding entry associated with the Sender IP address of the ARP packet does not match the port that the ARP packet was received on. ■ srcIpNotAllocated(4) indicates that the CHADDR of the DHCP Snooping binding entry associated with the Sender IP address of the ARP packet does not match the Source MAC and/or the ARP source MAC of the ARP packet.

AT-DNS-CLIENT-MIB

AT-DNS-CLIENT-MIB contains definitions of managed objects for the Allied Telesis DNS Client Configuration.

Objects in this group have the object identifier atDns ({ Modules 501 }). **Table 95-7** lists the objects supported by the AlliedWare Plus™ Operating System.

Table 95-7: Objects defined in AT-DNS-CLIENT-MIB

Object	Object Identifier	Description
atDnsClient	{ atDns 1 }	MIB File for DNS Client Configuration.
atDNSServerIndexNext	{ atDnsClient 1 }	The next available value for the object 'atDNSServerIndex'. The value is used by a management application to create an entry in the 'atDNSServerTable'.
atDNSServerTable	{ atDnsClient 2 }	Table of information about the Domain Name System (DNS) Server configurations in the system, indexed by 'atDNSServerIndex'.
atDNSServerEntry	{ atDNSServerTable 1 }	Information about a single DNS Server Configuration.
atDNSServerIndex	{ atDNSServerEntry 1 }	The index corresponding to the particular DNS Server Configuration. When creating a new entry in the table, the value of this object must be equal to the value in the 'atDNSServerIndexNext'.
atDNSServerAddrType	{ atDNSServerEntry 2 }	The Internet Address Type of the 'atDNSServerAddr' object. Can be one of the following: unknown (0) 1. ipv4 (1) - default 2. ipv6 (2) - not supported 3. ipv4z (3) - not supported 4. ipv6z (4) - not supported 5. dns (16) - not supported
atDNSServerAddr	{ atDNSServerEntry 3 }	The IP Address of the DNS Server. When a new entry is created, this object is set to the default of '0.0.0.0' { '00000000'h }. The management application will change this to the desired value using a SET operation.
atDNSServerStatus	{ atDNSServerEntry 4 }	The status of the current entry (row). Can be one of the following: 1. active (1) 2. createAndGo (4) 3. destroy (6) To create a new entry the management application must set this object with value 'createAndGo (4)'. To delete an entry, the management application must set this object with value 'destroy (6)'. Once an entry is deleted, all subsequent entries in the table will be renumbered. The default is 1 (active)

AT-ENVMONv2-MIB

The AT Environment Monitoring v2 MIB (atEnvMonv2-MIB) has the object path 207.8.4.4.3.12. It contains objects for managing and reporting data relating to fans, voltage rails, temperature sensors and power supply units installed in the device (**Table 95-8**). Objects in this group have the object identifier EnvMonv2 ({ sysinfo 12 }).

Table 95-8: Objects defined in AT-ENVMONV2-MIB

Object / Object Identifier	Description
atEnvMonv2Notifications { atEnvMonv2 0 } OID (207.8.4.4.3.12.0)	A collection of traps (notification) objects for monitoring fans, voltage rails, temperature sensors, and power supply bays.
atEnvMonv2FanAlarmSetNotify { atEnvMonv2Notifications 1 } OID (207.8.4.4.3.12.0.1)	A notification that is generated when the monitored speed of a fan drops below its lower threshold. It returns the value of: 1. atEnvMonv2FanStackMemberId 2. atEnvMonv2FanBoardIndex 3. atEnvMonv2FanIndex 4. atEnvMonv2FanDescription 5. atEnvMonv2FanLowerThreshold 6. atEnvMonv2FanCurrentSpeed
atEnvMonv2FanAlarmClearedNotify { atEnvMonv2Notifications 2 }	Notification generated when the monitored speed of a fan returns to an acceptable value, the fan having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2FanStackMemberId 2. atEnvMonv2FanBoardIndex 3. atEnvMonv2FanIndex 4. atEnvMonv2FanDescription 5. atEnvMonv2FanLowerThreshold 6. atEnvMonv2FanCurrentSpeed
atEnvMonv2VoltAlarmSetNotify { atEnvMonv2Notifications 3 }	Notification generated when the voltage of a monitored voltage rail, goes out of tolerance by either dropping below its lower threshold, or exceeding its upper threshold. It returns the value of: 1. atEnvMonv2VoltageStackMemberId 2. atEnvMonv2VoltageBoardIndex 3. atEnvMonv2VoltageIndex 4. atEnvMonv2VoltageDescription 5. atEnvMonv2VoltageUpperThreshold 6. atEnvMonv2VoltageLowerThreshold 7. atEnvMonv2VoltageCurrent (i.e. the voltage currently being measured).
atEnvMonv2VoltAlarmClearedNotify { atEnvMonv2Notifications 4 }	Notification generated when the voltage of a monitored voltage rail returns to an acceptable value, having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2VoltageStackMemberId 2. atEnvMonv2VoltageBoardIndex 3. atEnvMonv2VoltageIndex 4. atEnvMonv2VoltageDescription 5. atEnvMonv2VoltageUpperThreshold 6. atEnvMonv2VoltageLowerThreshold 7. atEnvMonv2VoltageCurrent (i.e. the voltage currently being measured).

Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2TempAlarmSetNotify { atEnvMonv2Notifications 5 }	Notification generated when a monitored temperature exceeds its upper threshold. It returns the value of: 1. atEnvMonv2TemperatureStackMemberId 2. atEnvMonv2TemperatureBoardIndex 3. atEnvMonv2TemperatureIndex 4. atEnvMonv2TemperatureDescription 5. atEnvMonv2TemperatureUpperThreshold 6. atEnvMonv2TemperatureCurrent
atEnvMonv2TempAlarmClearedNotify { atEnvMonv2Notifications 6 }	Notification generated when a monitored temperature returns to an acceptable value, having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2TemperatureStackMemberId 2. atEnvMonv2TemperatureBoardIndex 3. atEnvMonv2TemperatureIndex 4. atEnvMonv2TemperatureDescription 5. atEnvMonv2TemperatureUpperThreshold
atEnvMonv2PsbAlarmSetNotify { atEnvMonv2Notifications 7 }	Notification generated when a monitored parameter of a power supply bay device goes out of tolerance. It returns the value of: 1. atEnvMonv2PsbSensorStackMemberId 2. atEnvMonv2PsbSensorBoardIndex 3. atEnvMonv2PsbSensorIndex 4. atEnvMonv2PsbSensorType 5. atEnvMonv2PsbSensorDescription
atEnvMonv2PsbAlarmClearedNotify { atEnvMonv2Notifications 8 }	Notification generated when a monitored parameter of a power supply bay device returns to an acceptable value, having previously been in an alarm condition. It returns the value of: 1. atEnvMonv2PsbSensorStackMemberId 2. atEnvMonv2PsbSensorBoardIndex 3. atEnvMonv2PsbSensorIndex 4. atEnvMonv2PsbSensorType 5. atEnvMonv2PsbSensorDescription
atEnvMonv2ContactInputOpenNotify { atEnvMonv2Notifications 9 }	Notification generated when a monitored contact input opens. It returns the value of: 1. atEnvMonv2ContactInputStackMemberId 2. atEnvMonv2ContactInputBoardIndex 3. atEnvMonv2ContactInputIndex 4. atEnvMonv2ContactInputDescription
atEnvMonv2ContactInputCloseNotify { atEnvMonv2Notifications 10 }	Notification generated when a monitored contact input closes. It returns the value of: 1. atEnvMonv2ContactInputStackMemberId 2. atEnvMonv2ContactInputBoardIndex 3. atEnvMonv2ContactInputIndex 4. atEnvMonv2ContactInputDescription

Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2ExternalPSUAlarmSetNotify { atEnvMonv2Notifications 11 }	Notification generated when supply potential of a monitored external power supply is not present. It returns the value of: 1. atEnvMonv2ExternalPSUStackMemberId 2. atEnvMonv2ExternalPSUBoardIndex 3. atEnvMonv2ExternalPSUIndex 4. atEnvMonv2ExternalPSUDescription
atEnvMonv2ExternalPSUAlarmClearedNotify { atEnvMonv2Notifications 12 }	Notification generated when supply potential of a monitored external power supply returns to an acceptable level, having previously been in alarm condition. It returns the value of: 1. atEnvMonv2ExternalPSUStatusStackMemberId 2. atEnvMonv2ExternalPSUStatusBoardIndex 3. atEnvMonv2ExternalPSUStatusIndex 4. atEnvMonv2ExternalPSUStatusDescription
atEnvMonv2FanTable { EnvMonv2 1 } OID (207.8.4.4.3.12.1)	Table of information about fans installed in the device that have their fan speeds monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2FanStackMemberId 2. atEnvMonv2FanBoardIndex 3. atEnvMonv2FanIndex
atEnvMonv2FanEntry { atEnvMonv2FanTable 1 }	Description, current speed, lower threshold speed and current status of a single fan.
atEnvMonv2FanStackMemberId { atEnvMonv2FanEntry 1 }	Index of the stack member hosting this fan.
atEnvMonv2FanBoardIndex { atEnvMonv2FanEntry 2 }	Index of the board hosting this fan in the board table.
atEnvMonv2FanIndex { atEnvMonv2FanEntry 3 }	Numeric identifier of this fan on its host board.
atEnvMonv2FanDescription { atEnvMonv2FanEntry 4 }	Description of this fan.
atEnvMonv2FanCurrentSpeed { atEnvMonv2FanEntry 5 }	Current speed of this fan in revolutions per minute.
atEnvMonv2FanLowerThreshold { atEnvMonv2FanEntry 6 }	Minimum acceptable speed of the fan in revolutions per minute.
atEnvMonv2FanStatus { atEnvMonv2FanEntry 7 }	Whether this fan is currently in an alarm condition. The values can be: 1. Failed. Means that the current speed is too low. 2. Good. Means that the current speed is acceptable.
atEnvMonv2VoltageTable { atEnvMonv2 2 }	Table of information about voltage rails in the device that are monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2VoltageStackMemberId ■ atEnvMonv2VoltageBoardIndex ■ atEnvMonv2VoltageIndex
atEnvMonv2VoltageEntry { atEnvMonv2VoltageTable 1 }	Description, current value, upper & lower threshold settings and current status of a single voltage rail.
atEnvMonv2VoltageStackMemberId { atEnvMonv2VoltageEntry 1 }	Index of the stack member hosting this voltage sensor.

Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2VoltageBoardIndex { atEnvMonv2VoltageEntry 2 }	Index of the board hosting this voltage sensor in the board table.
atEnvMonv2VoltageIndex { atEnvMonv2VoltageEntry 3 }	Numeric identifier of this voltage rail on its host board.
atEnvMonv2VoltageDescription { atEnvMonv2VoltageEntry 4 }	Description of this voltage rail.
atEnvMonv2VoltageCurrent { atEnvMonv2VoltageEntry 5 }	Current reading of this voltage rail in millivolts.
atEnvMonv2VoltageUpperThreshold { atEnvMonv2VoltageEntry 6 }	Maximum acceptable reading of this voltage rail in millivolts.
atEnvMonv2VoltageLowerThreshold { atEnvMonv2VoltageEntry 7 }	Minimum acceptable reading of this voltage rail in millivolts.
atEnvMonv2VoltageStatus { atEnvMonv2VoltageEntry 8 }	Whether this voltage rail is currently in an alarm condition. Possible values are: 1. outOfRange (1) - means that the current reading is outside the threshold range. 2. inRange (2) - means that the current reading is acceptable.
atEnvMonv2TemperatureTable { atEnvMonv2 3 }	Table of information about temperature sensors in the device that are monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2TemperatureStackMemberId 2. atEnvMonv2TemperatureBoardIndex 3. atEnvMonv2TemperatureIndex 4. atEnvMonv2TemperatureDescription 5. atEnvMonv2TemperatureCurrent 6. atEnvMonv2TemperatureUpperThreshold 7. atEnvMonv2TemperatureStatus
atEnvMonv2TemperatureEntry { atEnvMonv2TemperatureTable 1 }	Description, current value, upper threshold setting and current status of a single temperature sensor.
atEnvMonv2TemperatureStackMemberId { atEnvMonv2TemperatureEntry 1 }	Index of the stack member hosting this temperature sensor.
atEnvMonv2TemperatureBoardIndex { atEnvMonv2TemperatureEntry 2 }	Index of the board hosting this temperature sensor in the board table.
atEnvMonv2TemperatureIndex { atEnvMonv2TemperatureEntry 3 }	Numeric identifier of this temperature sensor on its host board.
atEnvMonv2TemperatureDescription { atEnvMonv2TemperatureEntry 4 }	Description of this temperature sensor.
atEnvMonv2TemperatureCurrent { atEnvMonv2TemperatureEntry 5 }	Current reading of this temperature sensor in degrees Celsius.
atEnvMonv2TemperatureUpperThreshold { atEnvMonv2TemperatureEntry 6 }	Maximum acceptable reading for this temperature sensor in degrees Celsius.
atEnvMonv2TemperatureStatus { atEnvMonv2TemperatureEntry 7 }	Whether this temperature sensor is currently in an alarm condition. Can be: 1. outOfRange (1) - means that the current reading is outside the threshold range. 2. inRange (2) - means that the current reading is acceptable.

Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2PsbObjects { atEnvMonv2 4 }	Collection of objects for monitoring power supply bays in the system and any devices that are installed. It contains the following objects: 1. atEnvMonv2PsbTable ■ atEnvMonv2PsbSensorTable
atEnvMonv2PsbTable { atEnvMonv2PsbObjects 1 }	Table of information about power supply bays in the system, indexed by: 1. atEnvMonv2PsbHostStackMemberId 2. atEnvMonv2PsbHostBoardIndex 3. atEnvMonv2PsbHostSlotIndex 4. atEnvMonv2PsbHeldBoardIndex 5. atEnvMonv2PsbHeldBoardId 6. atEnvMonv2PsbDescription
atEnvMonv2PsbEntry { atEnvMonv2PsbTable 1 }	Description and current status of a single power supply bay device.
atEnvMonv2PsbHostStackMemberId { atEnvMonv2PsbEntry 1 }	Index of the stack member hosting this power supply bay.
atEnvMonv2PsbHostBoardIndex { atEnvMonv2PsbEntry 2 }	Index of the board hosting this power supply bay in the board table.
atEnvMonv2PsbHostSlotIndex { atEnvMonv2PsbEntry 3 }	Index of this power supply bay slot on its host board. This index is fixed for each slot, on each type of board.
atEnvMonv2PsbHeldBoardIndex { atEnvMonv2PsbEntry 4 }	Index of a board installed in this power supply bay. This value corresponds to atEnvMonv2PsbSensorBoardIndex for each sensor on this board. A value of 0 indicates that a board is either not present or not supported.
atEnvMonv2PsbHeldBoardId { atEnvMonv2PsbEntry 5 }	Type of board installed in this power supply bay. The values of this object are taken from the pprXxx object IDs under the boards sub-tree in the parent MIB. A value of 0 indicates that a board is either not present or not supported.
atEnvMonv2PsbDescription { atEnvMonv2PsbEntry 6 }	Description of this power supply bay.
atEnvMonv2PsbSensorTable { atEnvMonv2PsbObjects 2 }	Table of information about environment monitoring sensors on devices installed in power supply bays, indexed by: 1. atEnvMonv2PsbSensorStackMemberId ■ atEnvMonv2PsbSensorBoardIndex ■ atEnvMonv2PsbSensorIndex
atEnvMonv2PsbSensorEntry { atEnvMonv2PsbSensorTable 1 }	Description and current status of the sensor on a device installed in a power supply bay.
atEnvMonv2PsbSensorStackMemberId { atEnvMonv2PsbSensorEntry 1 }	Index of the stack member hosting this sensor.
atEnvMonv2PsbSensorBoardIndex { atEnvMonv2PsbSensorEntry 2 }	Index of the board hosting this sensor in the board table.
atEnvMonv2PsbSensorIndex { atEnvMonv2PsbSensorEntry 3 }	Index of this power supply bay environmental sensor on its host board.

Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2PsbSensorType { atEnvMonv2PsbSensorEntry 4 }	Type of environmental variable this sensor detects. One of: 1. psbSensorTypeInvalid(0) ■ fanSpeedDiscrete(1) ■ temperatureDiscrete(2) ■ voltageDiscrete(3)
atEnvMonv2PsbSensorDescription { atEnvMonv2PsbSensorEntry 5 }	Description of this power supply bay environmental sensor.
atEnvMonv2PsbSensorStatus { atEnvMonv2PsbSensorEntry 6 }	Whether this environmental sensor is currently in an alarm condition. One of: 1. failed (1) - the device is in a failure condition 2. good (2) - the device is functioning normally. 3. notPowered (3) - a PSU is installed, but not powered up
atEnvMonv2PsbSensorReading { atEnvMonv2PsbSensorEntry 7 }	An indication of whether this environmental sensor is currently reading a value for the monitored device. A value of 'no' indicates that there is no current reading, 'yes' indicates that the monitored device is supplying a reading. 1. no 2. yes
atEnvMonv2Traps { atEnvMonv2 5 } (207.8.4.4.3.12.5)	Note that objects under this portion of the tree have been deprecated, and replaced by objects under the tree portion 207.8.4.4.3.12.0.
atEnvMonv2FaultLedTable { atEnvMonv2 6 }	Table detailing any LED fault indications on the device, indexed by: 1. atEnvMonv2FaultLedStackMemberId
atEnvMonv2FaultLedEntry { atEnvMonv2FaultLedTable 1 }	Information pertaining to a given fault LED.
atEnvMonv2FaultLedStackMemberId { atEnvMonv2FaultLedEntry 1 }	Index of the stack member hosting this fault LED.
atEnvMonv2FaultLed1Flash { atEnvMonv2FaultLedEntry 2 }	Indicates whether a fault LED is currently showing a system failure by flashing once. Values can be: 1. heatsinkFanFailure (1) - indicates that one or more heatsink fans have failed, or are operating below the recommended speed 2. noFault (2)
atEnvMonv2FaultLed2Flashes { atEnvMonv2FaultLedEntry 3 }	Indicates whether a fault LED is currently showing a system failure by flashing twice. Values can be: 1. chassisFanFailure (1) - indicates that one or both of the chassis fans are not installed, or the fans are operating below the recommended speed 2. noFault (2)
atEnvMonv2FaultLed3Flashes { atEnvMonv2FaultLedEntry 4 }	Indicates whether a fault LED is currently showing a system failure by flashing three times. Values can be: 1. sensorFailure (1) - indicates that the ability to monitor temperature or fans has failed 2. noFault (2)
atEnvMonv2FaultLed4Flashes { atEnvMonv2FaultLedEntry 5 }	Indicates whether a fault LED is currently showing a system failure by flashing four times. Values can be: 1. xemInitialisationFailure (1) - indicates that a XEM failed to initialise or is incompatible 2. noFault (2)

Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2FaultLed5Flashes { atEnvMonv2FaultLedEntry 6 }	Indicates whether a fault LED is currently showing a system failure by flashing five times. Values can be: 1. alarmMonitorAlarm (1) - indicates that the Alarm Monitor has detected one or more fault conditions. 2. noFault (2)
atEnvMonv2FaultLed6Flashes { atEnvMonv2FaultLedEntry 7 }	Indicates whether a fault LED is currently showing a system failure by flashing six times. Values can be: 1. temperatureFailure (1) - indicates that the device's temperature has exceeded the recommended threshold 2. noFault (2)
atEnvMonv2ContactInputTable { atEnvMonv2 7 } (1.3.6.1.4.1.207.8.4.4.3.12.7)	Table of information about contact inputs available in the device that are monitored by environment monitoring hardware, indexed by: ■ atEnvMonv2ContactInputStackMemberId ■ atEnvMonv2ContactInputBoardIndex ■ atEnvMonv2ContactInputIndex
atEnvMonv2ContactInputEntry { atEnvMonv2ContactInputTable 1 }	The description and current state of a contact input.
atEnvMonv2ContactInputStackMemberId { atEnvMonv2ContactInputEntry 1 }	Index of the stack member hosting this input contact.
atEnvMonv2ContactInputBoardIndex { atEnvMonv2ContactInputEntry 2 }	Index of the board hosting this input contact in the board table.
atEnvMonv2ContactInputIndex { atEnvMonv2ContactInputEntry 3 }	The numeric identifier of this contact input on its host board.
atEnvMonv2ContactInputDescription { atEnvMonv2ContactInputEntry 4 }	The description of this contact input.
atEnvMonv2ContactInputState { atEnvMonv2ContactInputEntry 5 }	Current state of the input contact - closed (1) or open (2).
atEnvMonv2ContactIOutputTable { atEnvMonv2 8}	Table of information about contact outputs available in the device that are managed by environment monitoring hardware, indexed by: 1. atEnvMonv2ContactOutputStackMemberId 2. atEnvMonv2ContactOutputBoardIndex 3. atEnvMonv2ContactOutputIndex
atEnvMonv2ContactOutputEntry { atEnvMonv2ContactOutputTable 1 }	
atEnvMonv2ContactOutputStackMemberId { atEnvMonv2ContactInputEntry 1 }	Index of the stack member hosting this output contact.
atEnvMonv2ContactOutputBoardIndex { atEnvMonv2ContactInputEntry 2 }	Index of the board hosting this contact output in the board table.
atEnvMonv2ContactOutputIndex { atEnvMonv2ContactInputEntry 3 }	The numeric identifier of this contact output on its host board.
atEnvMonv2ContactOutputDescription { atEnvMonv2ContactInputEntry 4 }	The description of this contact output.
atEnvMonv2ContactOutputState { atEnvMonv2ContactInputEntry 5 }	Current state of the output contact - closed (1) or open (2).

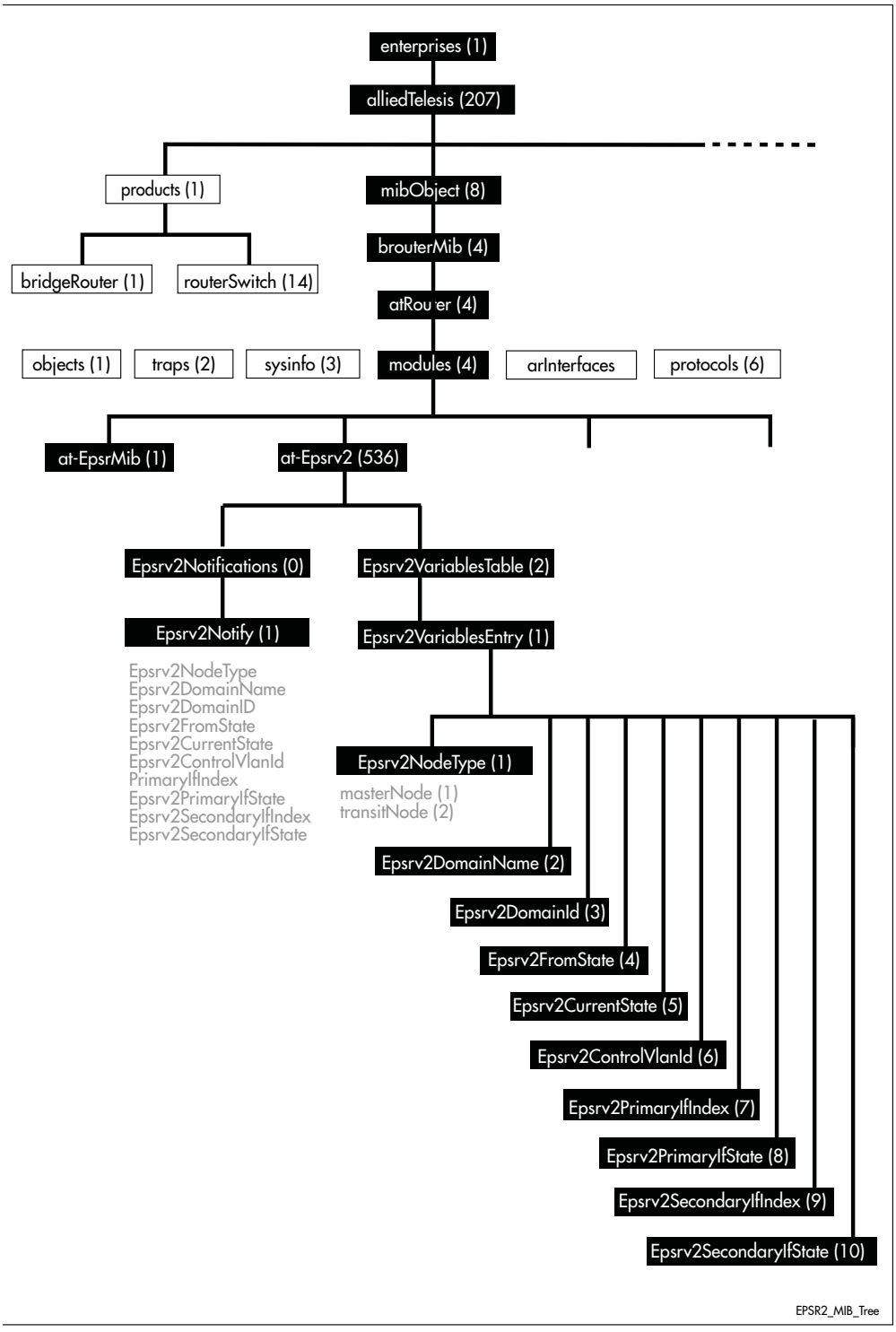
Table 95-8: Objects defined in AT-ENVMONV2-MIB(cont.)

Object / Object Identifier	Description
atEnvMonv2ExternalPSUTable { atEnvMonv2 9}	Table of information about external power supply status monitored by environment monitoring hardware, indexed by: 1. atEnvMonv2ExternalPSUStatusStackMemberId 2. atEnvMonv2ExternalPSUStatusBoardIndex 3. atEnvMonv2ExternalPSUStatusIndex
atEnvMonv2ExernalPSUEntry { atEnvMonv2ExternalPSUStatusTable 1 }	
atEnvMonv2ExternalPSUStackMemberId { atEnvMonv2ExternalPSUEntry 1 }	Index of the stack member hosting this external power supply.
atEnvMonv2ExternalPSUBoardIndex { atEnvMonv2ExternalPSUEntry 2 }	Index of the board hosting this external power supply in the board table.
atEnvMonv2ExternalPSUIndex { atEnvMonv2ExternalPSUEntry 3 }	The numeric identifier of this external power supply on its host board.
atEnvMonv2ExternalPSUDescription { atEnvMonv2ExternalPSUEntry 4 }	The description of this external power supply.
atEnvMonv2ExternalPSUState { atEnvMonv2ExternalPSUEntry 5 }	Current state of the output contact - good (1) or failed (2).

AT-EPSRv2-MIB

The EPSRv2 Group-MIB defines objects for managing Epsrv2 objects and triggers (**Figure 95-4**, **Table**). Objects in this group have the object identifier Epsrv2 ({ modules 536 }).

Figure 95-4: The AT-EPSRv2 MIB sub-tree



atEpsrv2Objects Defined in the AT-EPSRV2 MIB

Object	Object Identifier	Description
{ at-Epsrv2 }	{ modules 536 }	The root of the Epsrv2 object sub tree.
{ atEpsrv2Notifications }	{ at-Epsrv2 0 }	
{ atEpsrv2Notify }	{ atEpsrv2Notifications 1 }	EPSR Master/Transit node state transition trap. Note that there is a one to one relationship between nodes and domains.
{ Epsrv2NodeType }	{ atEpsrv2VariablesEntry 1 }	The EPSR node type: either master or transit.
{ atEpsrv2DomainName }	{ atEpsrv2VariablesEntry 2 }	The name of the EPSR domain.
{ atEpsrv2DomainID }	{ atEpsrv2VariablesEntry 3 }	The ID of the EPSR domain.
{ Epsrv2FromState }	{ atEpsrv2VariablesEntry 4 }	The previous state of the EPSR domain
{ Epsrv2Current State }	{ atEpsrv2VariablesEntry 5 }	The current state of the EPSR domain.
{ Epsrv2ControlVlanId }	{ atEpsrv2VariablesEntry 6 }	The VLAN identifier for the control VLAN.
{ Epsrv2PrimaryIfIndex }	{ atEpsrv2VariablesEntry 7 }	The IfIndex of the primary interface.
{ atEpsrv2PrimaryIfState }	{ atEpsrv2VariablesEntry 8 }	The current state of the primary interface.
{ atEpsrv2SecondaryIfIndex }	{ atEpsrv2VariablesEntry 9 }	The IfIndex of the secondary interface.
{ atEpsrv2SecondaryIfState }	{ atEpsrv2VariablesEntry 10 }	The state of the secondary interface.
{ atEpsrv2VariablesTable }	{ at-Epsrv2 2 }	The enterprise Epsrv2VariablesTable.
{ atEpsrv2VariablesEntry }	{ atEpsrv2VariablesTable 1}	Contains entries within the enterprise atEpsrv2VariablesTable.
{ atEpsrv2NodeType }	{ atEpsrv2VariablesEntry 1 }	The EPSR domain node type: either 1. master (1) 2. transit (2)
{ atEpsrv2DomainName }	{ Epsrv2NodeType 2 }	The name of the EPSR domain.
{ atEpsrv2DomainID }	{ Epsrv2NodeType 3 }	The ID of the EPSR domain.
{ atEpsrv2FromState }	{ Epsrv2NodeType 4 }	The previous state of the EPSR domain
{ atEpsrv2Current State }	{ Epsrv2NodeType 5 }	The current state of the EPSR domain.
{ atEpsrv2ControlVlanId }	{ Epsrv2NodeType 6 }	The VLAN identifier for the control VLAN.
{ Epsrv2PrimaryIfIndex }	{ Epsrv2NodeType 7 }	The IfIndex of the primary interface.
{ atEpsrv2PrimaryIfState }	{ Epsrv2NodeType 8 }	The current state of the primary interface.
{ atEpsrv2SecondaryIfIndex }	{ Epsrv2NodeType 9 }	The IfIndex of the secondary interface.
{ atEpsrv2SecondaryIfState }	{ Epsrv2NodeType 10 }	The state of the secondary interface.
TEXTUAL CONVENTIONS		
{ atEpsrv2NodeState }		The trap states that can be advertised for an EPSR domain node. The following states are defined: 1. idle (1) 2. complete (2) 3. failed (3) 4. linksUp (4) 5. linksDown (5) 6. preForward (6) 7. unknown (7)
{ atEpsrv2InterfaceState }		The trap states that can be advertised for an EPSR interface. The following states are defined: 1. unknown (1) 2. down (2) 3. blocked (3) 4. forward (4)

AT-FILEv2-MIB

This MIB contains objects for displaying and managing file content of Flash, USB storage devices and NVS, and copying, moving and deleting files from local and remote sources (**Table 95-9**).

The objects reside in the module atFilev2 { modules 600 }, organized in the following groups:

- The file operation devices - object for various devices supported for file operations
- The File Info Table - information about all files, including pathnames, that are present on the device
- The USB storage device table - information about the USB storage device configured on the device

The procedure in **"Copy a File to or from a TFTP Server" on page 93.20** shows how to use these MIB objects to upgrade to a new software version and boot configuration file.

Table 95-9: Objects defined in AT-FILEv2-MIB

Object	Object Identifier	Description
atFilev2	{ modules 600 }	MIB containing objects for listing and managing files.
atFilev2FileOperation	{ atFilev2 3 }	Collection of file operation objects available for configuration, to enable copying, moving and deleting files.
atFilev2SourceStackID	{ atFilev2Operation 1 }	Specifies the Stack ID of the source file. Set an integer corresponding to the stack ID of the stack member to use as the source. For devices that are not capable of being stacked, set with the value 1. This value is ignored if the source device is set to TFTP.
atFilev2SourceDevice	{ atFilev2Operation 2 }	Specifies the source device for the file to be copied. Valid values are 1 to 5. Set a value that corresponds with the various devices, as below: ■ 1 - Flash - default ■ 2 - Card - not supported ■ 3 - NVS ■ 4 - TFTP ■ 5 - USB For copying files, you may use any combination of devices for the source and destination, except for copying from TFTP to TFTP. For moving files you cannot use TFTP as source or destination. For deleting files, the source cannot be TFTP. You must fully configure all required parameters before an operation can commence. Where a TFTP operation is configured, an IP address must also be set via atFilev2TftpIPAddr. To copy a file from TFTP to Flash, use 4 for source and 1 for destination.

Table 95-9: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2SourceFilename	{ atFilev2Operation 3 }	Specifies the filename of the source file to copy, move or delete. Include any path as required, but the storage type is not necessary. For example, to copy the file `latest.cfg` from the backupconfigs/routers directory on the TFTP server, you would set: `backupconfigs/routers/latest.cfg`
atFilev2DestinationStackID	{ atFilev2Operation 4 }	Specifies the Stack ID for the destination file. For devices that are not capable of being stacked, set with the value 1. This value is ignored if the destination device is set to TFTP, or if a deletion operation is carried out.
atFilev2DestinationDevice	{ atFilev2Operation 5 }	Specifies the destination device for the files to be copied into. Valid values are 1 to 5. Set a value that corresponds with the various devices, as below: ■ 1 - Flash - default ■ 2 - Card - not supported ■ 3 - NVS ■ 4 - TFTP ■ 5 - USB For copying files, you may use any combination of devices for the source and destination, except for copying from TFTP to TFTP. For moving files you cannot use TFTP as source or destination. For deleting files, this object is ignored. You must fully configure all required parameters before an operation can commence. Where a TFTP operation is configured, an IP address must also be set via atFilev2TftpIPAddr. To copy a file from TFTP to Flash, use 4 for source and 1 for destination.
atFilev2DestinationFilename	{ atFilev2Operation 6 }	Specifies the destination filename of the file to be copied or moved. Include any path as required, but the storage type is not necessary. The destination filename does not need to be the same as the source filename, and this object is ignored for file deletion operations. For example, to copy a release file from the TFTP server to the backup release directory on Flash, you would set: `backuprelease/latest.rel` Note: If the destination is set to Flash, card or NVS, any file at the destination that shares the destination filename will be overwritten by a move or copy operation.

Table 95-9: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2CopyBegin	{ atFilev2Operation 7 }	Represents the status of the copy file operation, in the form of octet string. A read on this object can return several possible values, depending on the current status of the system and the various file operation objects: ■ idle - There is no file operation in progress and all required objects have been set correctly. Setting a '1' to this object will begin the file copy. ■ Error codes: [1-7] - A copy operation cannot be started until these errors are resolved. See below for key. ■ [action]ing x [--> y] - A file operation is currently in progress. You cannot start another operation while the object is returning this value. ■ [action] x [--> y] success - The last copy, move or delete operation was successfully completed. ■ [action] x [--> y] failure: [err] - The last copy, move or delete operation failed, with the error message attached. Common failures include lack of space on the destination file system, incorrect source file names or communication errors with remote services. Upon reading a success or failure message, the message will be cleared and the next read will result in either an 'idle' message or an 'Error codes' message if not all required objects have been correctly set. If the read returned 'idle', a new file operation can now be started. Following are possible values returned as Error codes for file copy: ■ 1 - atFilev2SourceDevice has not been set ■ 2 - atFilev2SourceFilename has not been set ■ 3 - atFilev2DestinationDevice has not been set ■ 4 - atFilev2DestinationFilename has not been set ■ 5 - atFilev2SourceDevice and atFilev2DestinationDevice are both set to TFTP ■ 6 - the combination of source device, stackID and filename is the same as the destination device, stackID and filename (i.e. it is not valid to copy a file onto itself. ■ 7 - TFTP IP address has not been set and TFTP has been set for one of the devices Provided all above requirements are met, immediately upon executing the SNMP set, the device will indicate that it was a success. The actual file copy itself will be started and continue on the device until it has completed. For large files, operations can take several minutes to complete. Subsequent reads of the object will return one of messages shown in the first table, to allow for tracking of the progress of the copy operation.

Table 95-9: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2MoveBegin	{ atFilev2Operation 8 }	Represents the status of the move file operation, in the form of octet string. A read on this object can return several possible values, depending on the current status of the system and the various file operation objects: ■ idle - There is no file operation in progress and all required objects have been set correctly. Setting a '1' to this object will begin the file move. ■ Error codes: [1-6] - A move operation cannot be started until these errors are resolved. See below for key. ■ [action]ing x [--> y] - A file operation is currently in progress. You cannot start another operation while the object is returning this value. ■ [action] x [--> y] success - The last copy, move or delete operation was successfully completed. ■ [action] x [--> y] failure: [err] - The last copy, move or delete operation failed, with the error message attached. Common failures include lack of space on the destination file system, incorrect source file names or communication errors with remote services. Upon reading a success or failure message, the message will be cleared and the next read will result in either an 'idle' message or an 'Error codes' message if not all required objects have been correctly set. If the read returned 'idle', a new file operation can now be started. Following are possible values returned as Error codes for file move: ■ 1 - atFilev2SourceDevice has not been set ■ 2 - atFilev2SourceFilename has not been set ■ 3 - atFilev2DestinationDevice has not been set ■ 4 - atFilev2DestinationFilename has not been set ■ 5 - either atFilev2SourceDevice or atFilev2DestinationDevice are set to TFTP ■ 6 - the combination of source device, stackID and filename is the same as the destination device, stackID and filename (i.e. it is not valid to move a file onto itself. Provided all above requirements are met, immediately upon executing the SNMP set, the device will indicate that it was a success. The actual file move itself will be started and continue on the device until it has completed. For large files, operations can take several minutes to complete. Subsequent reads of the object will return one of messages shown in the first table, to allow for tracking of the progress of the move operation.

Table 95-9: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2DeleteBegin	{ atFilev2Operation 9 }	Represents the status of the delete file operation, in the form of octet string. A read on this object can return several possible values, depending on the current status of the system and the various file operation objects: ■ idle - There is no file operation in progress and all required objects have been set correctly. Setting a '1' to this object will begin the file deletion. ■ Error codes: [1-3] - A delete operation cannot be started until these errors are resolved. See below for key. ■ [action]ing x [--> y] - A file operation is currently in progress. You cannot start another operation while the object is returning this value. ■ [action] x [--> y] success - The last copy, move or delete operation was successfully completed. ■ [action] x [--> y] failure: [err] - The last copy, move or delete operation failed, with the error message attached. Common failures include lack of space on the destination file system, incorrect source file names or communication errors with remote services. Upon reading a success or failure message, the message will be cleared and the next read will result in either an 'idle' message or an 'Error codes' message if not all required objects have been correctly set. If the read returned 'idle', a new file operation can be started. File deletion operations ignore the values set in the atFilev2DestinationStackID, atFilev2DestinationDevice and atFilev2DestinationFilename objects. The file deletion operation is equivalent to the CLI 'delete force [file]' command, so it is possible to delete any normally-protected system files, such as the currently configured boot release. Following are possible values returned as Error codes for file move: ■ 1 - atFilev2SourceDevice has not been set ■ 2 - atFilev2SourceFilename has not been set ■ 3 - atFilev2SourceDevicehas not been set to TFTP Provided all above requirements are met, immediately upon executing the SNMP set, the device will indicate that it was a success. The actual file move itself will be started and continue on the device until it has completed. For large files, operations can take several minutes to complete. Subsequent reads of the object will return one of messages shown in the first table, to allow for tracking of the progress of the move operation.
atFilev2Flash_1	{ atFilev2Operation 10 }	Represents the Flash operation device object
atFilev2Card_2	{ atFilev2Operation 11 }	Represents the Card operation device object
atFilev2Nvs_3	{ atFilev2Operation 12 }	Represents the NVS operation device object
atFilev2Tftp_4	{ atFilev2Operation 13 }	Represents the TFTP operation device object
atFilev2TftpIPAddr	{ atFilev2Tftp_4 1 }	The IP address of the TFTP server that is to be used for the file copy process. This IP Address needs to be reachable from the device, or the file copy will fail.

Table 95-9: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2Usb	{ atFilev2Operation 15 }	Represents the USB storage device operation device object.
atFilev2InfoEntry	{ atFilev2InfoTable 1 }	An entry in the list of files, containing information about a single file.
atFilev2InfoFilepath	{ atFilev2InfoEntry 1 }	The full path and name of the file. Files are sorted in alphabetical order and any filepath that is longer than 112 characters will not be displayed due to SNMP Object Identifier length limitations.
atFilev2InfoFileSize	{ atFilev2InfoEntry 2 }	The size of the file in bytes.
atFilev2InfoFileCreationTime	{ atFilev2InfoEntry 3 }	File creation time in the form <MMM DD YYYY HH:MM:SS>. For example, Sep 7 2008 06:07:54.
atFilev2InfoFileIsDirectory	{ atFilev2InfoEntry 4 }	This object will return the value TRUE if the entry is a directory, or FALSE if it is not.
atFilev2InfoFileIsReadable	{ atFilev2InfoEntry 5 }	This object will return the value TRUE if the file is readable, or FALSE if it is not.
atFilev2InfoFileIsWriteable	{ atFilev2InfoEntry 6 }	This object will return the value TRUE if the file is writeable, or FALSE if it is not.
atFilev2InfoFileIsExecutable	{ atFilev2InfoEntry 7 }	This object will return the value TRUE if the file is executable, or FALSE if it is not.
atFilev2USBMediaTable	{ atFilev2 6 }	The USB storage device table, containing information related to USB storage devices.
atFilev2USBMediaEntry	{ atFilev2USBMediaTable 1 }	Data pertaining to a USB storage device instance.
atFilev2USBMediaStackMemberId	{ atFilev2USBMediaEntry 1 }	The index of the stack member hosting this USB media. For devices that are not capable of being stacked, this object will always return the value 1.
atFilev2USBMediaPresence	{ atFilev2USBMediaEntry 2 }	This object indicates whether or not a USB storage device is inserted in a slot. Possible values are: ■ notPresent (1) ■ present (2)
atFilev2FileViewerStackId	atFilev2FileViewer 1	The stack ID of the stack member for which files will be displayed in the FileViewer table. For devices that are not capable of being stacked, this variable will always read as 1, and will cause an error on being written to with any value other than 1. Write this variable with the stack ID of the stack member for which a view of files is required. If the stack member doesn't exist, an error will be returned. For a chassis switch, it corresponds to the card ID. Note that the other variables specifying the files to view will not be altered by changing the stack ID, which means that the file view table could be empty if a non-existant device or path has been referenced previously.

Table 95-9: Objects defined in AT-FILEv2-MIB(cont.)

Object(cont.)	Object Identifier	Description
atFilev2FileViewerDevice	atFilev2FileViewer 2	The file system device for which files will be displayed in the FileViewer table. The values supported for this variable are identical to the values for other variables in the MIB, although not all values will actually result in the display of files. The different devices and whether they will result in the display of files are: ■ 1 - Flash - Onboard Flash - supported ■ 2 - Card - Removable SD card - supported ■ 3 - NVS - Onboard battery backed RAM - supported ■ 4 - FTP - not supported ■ 5 - USB - Removable USB media - supported Setting this variable to a unsupported value will result in an error, but setting to a value that is supported but on a device that doesn't contain that type of device will not. However, no files will be displayed in the File Viewer table in this case.
atFilev2FileViewerCurrentPath	atFilev2FileViewer 3	The file system path for which files will be displayed in the FileViewer table. This path will always read as a full pathname starting with the '/' character. Setting this variable will specify a new directory for which files will be displayed. The path specified must be the full path, relative setting of path does not work. Only paths with invalid characters in them will cause an error, paths specifying non-existent directories will be accepted, but no files will be displayed in the File Viewer table in this case.
atFilev2FileViewerTable	atFilev2FileViewer 4	A list of all files, not including pathnames, that are present on the device specified by atFilev2FileViewerStackId and atFilev2FileViewerDevice, in the path specified by atFilev2FileViewerCurrentPath. Hidden and system files are not displayed. If the Stack ID, device and path are invalid (the path is for a non-existent directory), the table will be empty. This will allow an MIB walk through the table even though the setup parameters are incorrect.
atFilev2FileViewerEntry	atFilev2FileViewerTable 1	An entry in the list of files, containing information about a single file.
atFilev2FileViewerName	atFilev2FileViewerEntry 1	The name of the file. Files are sorted in alphabetical order, and any name that is longer than 112 characters will not be displayed due to SNMP OID length limitations.
atFilev2FileViewerSize	atFilev2FileViewerEntry 2	The size of the file in bytes.
atFilev2FileViewerCreationTime	atFilev2FileViewerEntry 3	File creation time in the form <MMM DD YYYY HH:MM:SS>. For example, Sep 7 2008 06:07:54.
atFilev2FileViewerIsDirectory	atFilev2FileViewerEntry 4	Returns TRUE if the entry is a directory, FALSE otherwise.
atFilev2FileViewerIsReadable	atFilev2FileViewerEntry 5	Returns TRUE if the file is readable, FALSE otherwise.
atFilev2FileViewerIsWriteable	atFilev2FileViewerEntry 6	Returns TRUE if the file is writeable, FALSE otherwise.
atFilev2FileViewerIsExecutable	atFilev2FileViewerEntry 7	Returns TRUE if the file is executable, FALSE otherwise.

AT-IP-MIB

This MIB contains objects for Allied Telesis specific IP address management (**Table 95-10**). The objects reside in the module atIpMib { modules 602 }.

Table 95-10: Objects defined in AT-IP-MIB

Object	Object Identifier	Description
atIpMib	{ modules 602 }	MIB containing objects for IP addressing management.
AtIpAddressAssignmentType	Textual Convention	Object containing conditional coded values for the IP address assignment type being applied to the interface, referred to by objects in this MIB. The possible values and explanation are: ■ notSet (0) - indicates that the IP address assignment type has not yet been configured. This value can only ever be read. ■ primary (1) - indicates that the address is a primary IP address; only one primary address is allowed per interface. ■ secondary (2) - indicates that the address is a secondary IP address; any number of secondary IP addresses may be applied
AtIpAddressTable	{ atIpMib 1 }	A table containing mappings between primary or secondary IP addresses, and the interfaces they are assigned to. Indexed by: ■ atIpAddressAddrType ■ atIpAddressAddr
AtIpAddressEntry	{ AtIpAddressTable 1 }	Information about the address mapping for a particular interface.
atIpAddressAddrType	{ AtIpAddressEntry 1 }	An indication of the IP version of 'atIpAddressAddr'
atIpAddressAddr	{ AtIpAddressEntry 2 }	The IP address to which this entry's addressing information pertains. The address type of this object is specified in object 'atIpAddressAddrType'.
atIpAddressPrefixLen	{ AtIpAddressEntry 3 }	An integer, specifying the prefix length of the IP address represented by this entry.
atIpAddressLabel	{ AtIpAddressEntry 4 }	The name assigned to the IP address represented by this entry.
atIpAddressIfIndex	{ AtIpAddressEntry 5 }	The index value that uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index corresponds to the interface identified by the same value of the IF-MIB's ifIndex.
atIpAddressAssignmentType	{ AtIpAddressEntry 6 }	The IP address assignment type for this entry (primary or secondary), as described in the Textual Convention 'AtIpAddressAssignmentType'.

Table 95-10: Objects defined in AT-IP-MIB(cont.)

Object	Object Identifier	Description
atIpAddressRowStatus	{ AtIpAddressEntry 7 }	The current status of the IP address entry. The following values may be returned when reading this object: ■ active (1) « The IP address is currently mapped to an interface and is valid. ■ notReady (3) « The IP address is currently partially configured and is not mapped to an interface. The following values may be written to this object: ■ active (1) « An attempt will be made to map the IP address to the configured interface. ■ createAndWait (5) « An attempt will be made to create a new IP address entry. ■ destroy (6) « The IP address setting will be removed from the device. An entry cannot be made active until its atIpAddressPrefixLen, atIpAddressIfIndex and atIpAddressAssignmentType objects have been set to valid values.

AT-LICENSE-MIB

The AT-LICENSE-MIB contains objects for managing the AlliedWare Plus™ Operating System software licenses: listing applied software licenses, adding new licenses and deleting existing licenses (**Table 95-11**). The objects reside in the module license { sysinfo 22 }, organized in the following groups:

- Base Software License Table - a table containing the installed base software licenses on the device
- Installed Software License Table - a list of installed software licenses; used also to remove software license from the device
- Available Software Features Table
- LicenseNew - Objects used to install a new license
- LicenseStackRemove - Objects used to remove a license across a stack of devices

Table 95-11: Objects defined in AT-LICENSE-MIB

Object	Object Identifier	Description
license	{ sysinfo 22 }	MIB containing objects for listing applied software licenses, adding new licenses, and deleting existing licenses.
baseLicenseTable	{ license 1 }	Table containing information about base software licenses installed on a device. Indexed by: ■ baseLicenseStkId
baseLicenseEntry	{ baseLicenseTable 1 }	Information about a single license installed on the device.
baseLicenseStkId	{ baseLicenseEntry 1 }	The stack member ID of the device hosting the license.
baseLicenseName	{ baseLicenseEntry 2 }	The name of the base license.
baseLicenseQuantity	{ baseLicenseEntry 3 }	The number of licenses issued for this entry.
baseLicenseType	{ baseLicenseEntry 4 }	The type of base license issued.
baseLicenseIssueDate	{ baseLicenseEntry 5 }	The date of issue of the base license.
baseLicenseExpiryDate	{ baseLicenseEntry 6 }	The expiry date of the base license.
baseLicenseFeatures	{ baseLicenseEntry 7 }	The feature set that this license enables, in the format of an octet string. Each bit in the returned octet string represents a particular feature that can be license-enabled. The bit position within the string maps to the feature entry with the same index, in licenseFeatureTable. A binary '1' indicates that the feature is included in the license; a binary '0' indicates that the feature is not included in the license.
licenseTable	{ license 2 }	Table containing information about software licenses installed on the device. Indexed by: ■ licenseStackId ■ licenseIndex
licenseEntry	{ licenseTable 1 }	Information about a single installed software license on the device.
licenseStackId	{ licenseEntry 1 }	The stack member ID of the device hosting the license.
licenseIndex	{ licenseEntry 2 }	The index number of the license entry.
licenseName	{ licenseEntry 3 }	The name of the license.
licenseCustomer	{ licenseEntry 4 }	The name of the customer of the license.

Table 95-11: Objects defined in AT-LICENSE-MIB(cont.)

Object	Object Identifier	Description
licenseQuantity	{ licenseEntry 5 }	The number of licenses issued for this entry.
licenseType	{ licenseEntry 6 }	The type of license issued.
licenseIssueDate	{ licenseEntry 7 }	The date of issue of the license.
licenseExpiryDate	{ licenseEntry 8 }	The expiry date of the license.
licenseFeatures	{ licenseEntry 9 }	The feature set that this license enables, in the format of octet string. Each bit in the returned octet string represents a particular feature that can be license-enabled. The bit position within the string maps to the feature entry with the same index, in licenseFeatureTable. A binary '1' indicates that the feature is included in the license; a binary '0' indicates that the feature is not included in the license.
licenseRowStatus	{ licenseEntry 10 }	The current status of the license. The following values may be returned when reading this object: 1. active (1) - the license is currently installed and valid 2. notInService (2) - the license has expired or is invalid The following value may be written to this object: 3. destroy (6) - the license will be removed from the device; this may result in some features being disabled. Note that a stacked device that has a license deleted may not be able to rejoin the stack after reboot, unless the license is also deleted on all other devices in the stack.
licenseFeatureTable	{ license 3 }	Table containing all available Software Features. A feature must be license-enabled to be utilized on the device.
licenseFeatureEntry	{ licenseFeatureTable 1 }	Information about a single feature that must be license-enabled in order to be utilized on the device.
licenseFeatureIndex	{ licenseFeatureEntry 1 }	The index number of the feature which must be license-enabled.
licenseFeatureName	{ licenseFeatureEntry 2 }	The name of the feature under licensing control.
licenseFeatureStkMembers	{ licenseFeatureEntry 3 }	The set of stack members on which the feature is enabled, in the format of an octet string. Each bit in the string maps to an individual stacking member, e.g. bit one represents stacking member one, bit two represents stacking member two. A bit value of '1' indicates that the applicable feature is enabled on the matching device; a bit value of '0' indicates that the feature is disabled.
licenseNew	{ license 4 }	Group of objects available for updates, used when installing a new software license on the device.
licenseNewStackId	{ licenseNew 1 }	The ID of the stacking member upon which the new license is to be installed. The value zero (0) indicates that the license should be applied to all stack members.
licenseNewName	{ licenseNew 2 }	The name of the new license to be installed.
licenseNewKey	{ licenseNew 3 }	The key for the new license to be installed.

Table 95-11: Objects defined in AT-LICENSE-MIB(cont.)

Object	Object Identifier	Description
licenseNewInstall	{ licenseNew 4 }	Used to install new licenses. Values can be: 1. true (1) 2. false (2) To commence installation, a valid license name and key must first have been set via the licenseNewName and licenseNewKey respectively. This object should then be set to the value true (1). If either the license name or key is invalid, the write operation will fail. Once installed, the software modules affected by any newly enabled features will automatically be restarted. Note that a stacked device that has a new license installed on it may not be able to rejoin the stack after reboot, unless the license is also added to all other devices in the stack. When read, the object will always return the value false (2).
licenseNewInstallStatus	{ licenseNew 5 }	The current status of the last license installation request. One of the following values is returned when reading this object: ■ idle (1) ■ processing (2) ■ success (3) ■ failed (4) When a stack license installation operation is complete the first read of this object will return either a success (3) or a failure (4) indication. Subsequent reads of this object will then return an idle (1) indication.
licenseStackRemove	{ license 5 }	Group of objects used when removing a software license across a stack of devices.
licenseStackRemoveName	{ licenseStackRemove 1 }	The name of the license to be removed from all devices across the stack, on which the license currently exists.
licenseStackRemoveExecute	{ licenseStackRemove 2 }	When set to the value true (1), the system will attempt to remove the named license from all devices across the stack on which the license currently exists. All devices in a stack must be from the same product family and the named license must activate the same feature set on all devices.
licenseStackRemoveStatus	{ licenseStackRemove 3 }	The current status of the last requested stack license removal request. One of the following values is returned when reading this object: ■ idle (1) ■ processing (2) ■ success (3) ■ failed (4) When a stack license removal operation is complete the first read of this object will return either a success (3) or failure (4) indication. Subsequent reads of this object will then return an idle (1) indication.

AT-LOG-MIB

The AT Log MIB contains objects for listing log entries from the buffered and permanent logs (**Table 95-12**). The objects reside in the module log { modules 601 }, organized in the following groups:

- Log Table - objects containing the information from log messages issued by the system, ordered from oldest to newest entry
- Log Options - contains objects used to set up the log options configuration

Table 95-12: Objects defined in AT-LOG-MIB

Object	Object Identifier	Description
log	{ modules 601 }	MIB containing objects for listing log entries from the buffered and permanent logs.
logTable	{ log 1 }	A list of log entries from the source specified in the 'logSource' object. The list is ordered from oldest entry to newest entry. Indexed by: ■ logIndex
logEntry	{ logTable 1 }	Information about a single log entry, from the source specified in the 'logSource' object.
logIndex	{ logEntry 1 }	An index integer. This index is not directly tied to any specific log entry. Over time, the log will grow larger and eventually older entries will be removed from the log.
logDate	{ logEntry 2 }	The date of the log entry. Data resides in the format octet string, in the form YYYY MMM DD, e.g. 2008 Oct 9.
logTime	{ logEntry 3 }	The time of the log entry. Data resides in the format octet string, in the form HH:MM:SS, e.g. 07:15:04.
logFacility	{ logEntry 4 }	The syslog facility that generated the log entry, in the format octet string. See the reference manual for more information.
logSeverity	{ logEntry 5 }	The severity level of the log entry, in the format octet string. Severities are given below: ■ emerg Emergency, system is unusable ■ alert Action must be taken immediately ■ crit Critical conditions ■ errr Error conditions ■ warning Warning conditions ■ notice Normal, but significant, conditions ■ info Informational messages ■ debug Debug-level messages
logProgram	{ logEntry 6 }	The program that generated the log entry, in the format octet string. See the reference manual for more information.
logMessage	{ logEntry 7 }	The message of thew log entry, in the format octet string.
logOptions	{ log 2 }	Contains objects used to set up the required log options configuration.

Table 95-12: Objects defined in AT-LOG-MIB(cont.)

Object	Object Identifier	Description
logSource	{ logOptions 1 }	An integer indicating the source from which the log entries are retrieved. The valid values are: ■ 1 - Buffered log (default) ■ 2 - Permanent log. This information is used when retrieving the logTable objects, and also specifies the log to be cleared when the 'clearLog' object is set.
logAll	{ logOptions 2 }	An integer indicating whether to display all log entries in the logTable objects, or not. The valid values are: ■ 0 - to display only the most recent log messages. This is the default ■ 1 - to show all available log entries. Note: Choosing to display all log entries may result in delays of several seconds when accessing the logTable objects.
clearLog	{ logOptions 3 }	An integer indicating whether to clear the log that is specified by the 'logSource' object. Valid values are: ■ 0 - do not clear log ■ 1 - clear log

AT-LOOPPROTECT-MIB

The atLoopProtect-MIB (**Figure 95-5**, **Table 95-13**) defines objects for managing Loop Protection objects and triggers. Objects in this group have the object identifier atLoopProtect ({ modules 4 }).

Figure 95-5: The ATLoopProtect MIB Sub-tree

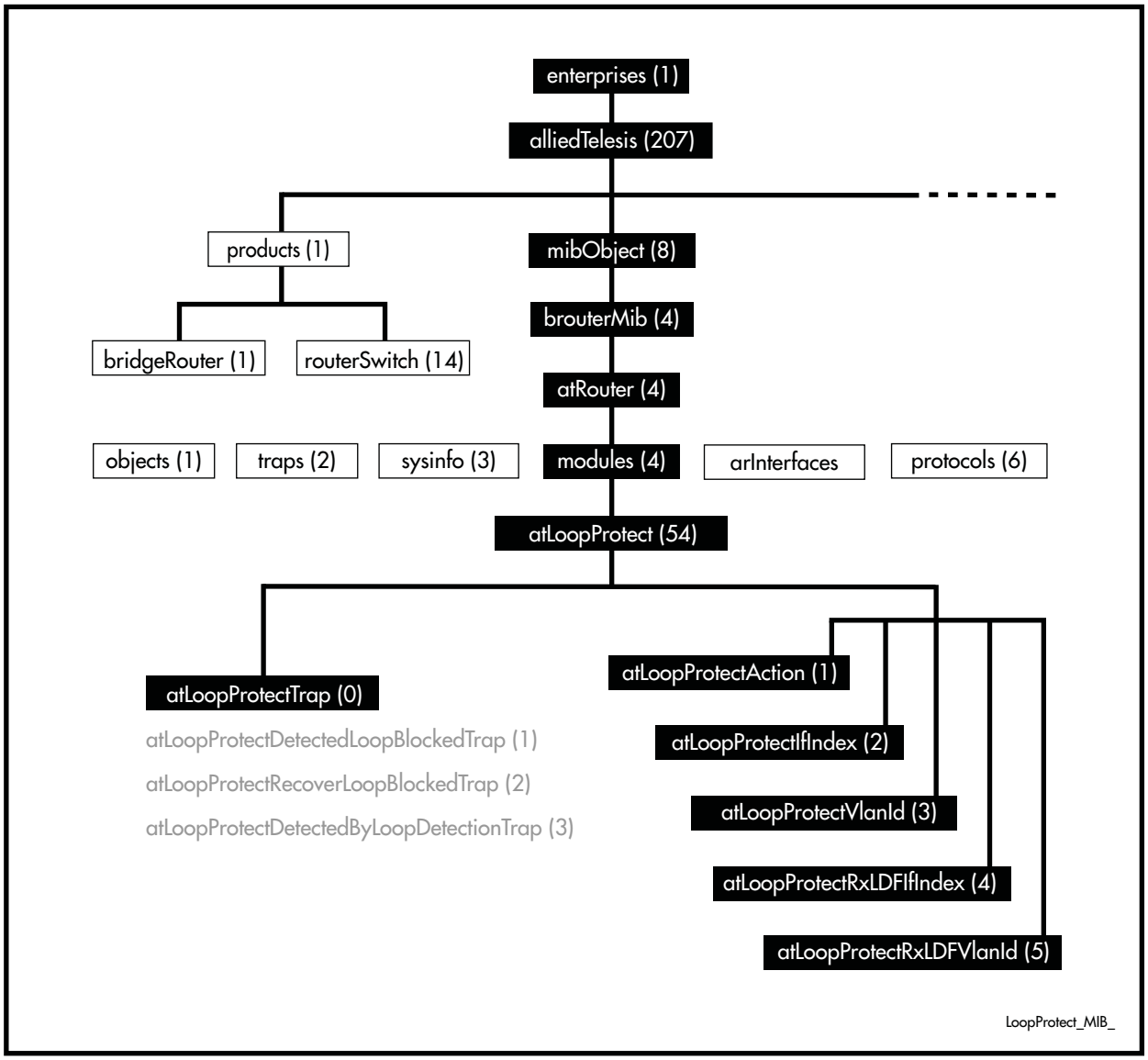


Table 95-13: Objects Defined in the AT-Loop Protect MIB

Object	Object Identifier	Description
{ atLoopProtect }	{ modules 54 }	The root of the Loop Protect object sub tree.
{ atLoopProtectTrap }	{ atLoopProtect0 }	The Loop Protection node state transition trap. List of traps (notifications) generated for Loop Protection.
{ atLoopProtectDetected LoopBlockedTrap }	{ atLoopProtectTrap1 }	Notification generated when the Loop Protection feature blocks an interface with a loop. The following bindings are associated with this trap: 1. atLoopProtectIfIndex 2. atLoopProtectVlanId 3. atLoopProtectAction
{ atLoopProtectRecover LoopBlockedTrap }	{ atLoopProtectTrap2 }	Notification generated when the Loop Protection feature restores a blocked interface back to normal operation. The following bindings are associated with this trap: 1. atLoopProtectIfIndex 2. atLoopProtectVlanId 3. atLoopProtectAction
{ atLoopProtectDetected ByLoopDetectionTrap }	{ atLoopProtectTrap3 }	Notification generated when the Loop Protection feature detects a loop by Loop Detection method. The following bindings are associated with this trap: 1. atLoopProtectIfIndex 2. atLoopProtectVlanId 3. atLoopProtectRxLDFIfIndex 4. atLoopProtectRxLDFVlanId
{ atLoopProtectAction }	{ atLoopProtect1 }	The Action for the Loop Protection feature. The following values are defined: 1. atLoopProtectAction-LearnDisable (0) 2. atLoopProtectAction-LearnEnable (1) 3. atLoopProtectAction-PortDisable (2) 4. atLoopProtectAction-PortEnable (3) 5. atLoopProtectAction-LinkDown (4) 6. atLoopProtectAction-LinkUp (5) 7. atLoopProtectAction-VlanDisable (6) 8. atLoopProtectAction-VlanEnable (7)
{ atLoopProtectIfIndex }	{ atLoopProtect2 }	The interface on which the loop was detected.
{ atLoopProtectVlanId }	{ atLoopProtect3 }	The VLAN ID on which the loop was detected.
{ atLoopProtectRxLDFIfIndex }	{ atLoopProtect4 }	The interface on which the loop detection frame was received.
{ atLoopProtectRxLDFVlanId }	{ atLoopProtect5 }	The VLAN ID on which the loop detection frame was received.

AT-MIBVERSION-MIB

The AT-MIBVERSION-MIB contains an object to display the last software release that contained changes to the supported AT Enterprise MIB definition files (**Table 95-14**). Objects in this group have the object identifier atMibsetVersion ({ sysinfo 15 }).

Table 95-14: Object defined in AT-MIBVERSION-MIB

Object	Object Identifier	Description
atMibVersion	{ sysinfo 15 }	This object returns a five digit integer which indicates the last software release that contained changes to the supported AT Enterprise MIB definition files. For example, If the currently loaded software release on the device is 5.3.1-0.3 but the Enterprise MIBs have not changed since 5.3.1-0.1, then the value returned will be 53101.

AT-NTP-MIB

This MIB contains objects for managing the Allied Telesis Network Time Protocol (NTP) configuration (**Table 95-15**). The objects reside in the module atNtp { modules 502 }, organized in the following groups:

- NTP Peer/Server Table - a table containing information on the Network Time Protocol (NTP) peers or server configurations in the system.
- Associations Table - a list of installed software; used also to remove software from the device.
- Status Table - Objects in this group are not supported.

Table 95-15: Objects defined in AT-NTP-MIB

Object	Object Identifier	Description
atNtp	{ modules 502 }	MIB containing objects for configuring NTP.
atNtpPeerIndexNext	{ atNtp 6 }	The next available index number to be used for object 'atNtpPeerIndex'.
atNtpPeerTable	{ atNtp 7 }	Table containing information on the Network Time Protocol (NTP) peers or server configurations in the system. Indexed by: ■ atNtpPeerIndex
atNtpPeerEntry	{ atNtpPeerTable 1 }	Information about a single NTP server or peer configuration.
atNtpPeerIndex	{ atNtpPeerEntry 1 }	The index number corresponding to a particular NTP server or peer configuration in the system. To create a new entry, the value of this object should be the same as that of the value of atNtpPeerIndexNext object, otherwise the entry creation will fail.
atNtpPeerNameAddr	{ atNtpPeerEntry 2 }	The host name, or the IP address of the NTP peer. When a new row (entry) is created, this object is set with a default of '0.0.0.0', and the management application should change it to a desired value by using a SET operation.
atNtpPeerMode	{ atNtpPeerEntry 3 }	The mode of the peer. Can be one of the following: ■ server (1) ■ peer (2) - default
atNtpPeerPreference	{ atNtpPeerEntry 4 }	The values in this object specifies whether this peer is the preferred one. Valid values are 0 to 2: ■ 0 - unknown - default ■ 1 - not preferred ■ 2 - preferred When the value is 'not preferred' (1) NTP chooses the peer with which to synchronize the time on the local system. If the object is set to 'preferred' (2) NTP will choose the corresponding peer to synchronize the time with.

Table 95-15: Objects defined in AT-NTP-MIB(cont.)

Object	Object Identifier	Description
atNtpPeerVersion	{ atNtpPeerEntry 5 }	The NTP version the peer supports. Can be one of the following: ■ 0 - unknown - default ■ 1 - version 1 ■ 2 - version 2 ■ 3 - version 3 ■ 4 - version 4
atNtpPeerKeyNumber	{ atNtpPeerEntry 6 }	The authentication key number. Default number is 0.
atNtpPeerRow Status	{ atNtpPeerEntry 7 }	The current status of this peer entry. The following values may be returned when reading this object: ■ active (1) - this value is returned on reading of this entry. ■ createAndGo (4) - this value is set by the management application when creating a new entry ■ destroy (6) - value set by the management application when deleting the entry. When an entry is deleted, all subsequent entries in the table will be re-indexed.
atNtpAssociationTable	{ atNtp 10 }	Table containing information on the Network Time Protocol (NTP) associations. Indexed by: ■ atNtpAssociationIndex
atNtpAssociationEntry	{ atNtpAssociationTable 1 }	Information about a single NTP server or peer configuration.
atNtpAssociationIndex	{ atNtpAssociationEntry 1 }	The index number corresponding to a particular NTP server or peer configuration in the system. To create a new entry, the value of this object should be the same as that of the value of atNtpPeerIndexNext object, otherwise the entry creation will fail.
atNtpAssociationPeerAddr	{ atNtpAssociationEntry 2 }	The host name, or the IP address of the NTP peer. When a new row (entry) is created, this object is set with a default of '0.0.0.0', and the management application should change it to a desired value by using a SET operation.
atNtpAssociationStatus	{ atNtpAssociationEntry 3 }	The status of this association. Can be one of the following: ■ master (synced) ■ master (unsynced) ■ selected ■ candidate ■ configured ■ unknown
atNtpAssociationConfigured	{ atNtpAssociationEntry 4 }	The value in this object specifies whether the association is from configuration or not. Value can be: ■ configured ■ dynamic
atNtpAssociationRefClkAddr	{ atNtpAssociationEntry 5 }	The IP Address for the reference clock.
atNtpAssociationStratum	{ atNtpAssociationEntry 6 }	The stratum of the peer clock.

Table 95-15: Objects defined in AT-NTP-MIB(cont.)

Object	Object Identifier	Description
atNtpAssociationPoll	{ atNtpAssociationEntry 7 }	The time between NTP requests from the device to the server, in seconds.
atNtpAssociationReach	{ atNtpAssociationEntry 8 }	An integer that indicates the reachability status of the peer.
atNtpAssociationDelay	{ atNtpAssociationEntry 9 }	The round trip delay between the device and the server.
atNtpAssociationOffset	{ atNtpAssociationEntry 10 }	The difference between the device clock and the server clock.
atNtpAssociationDisp	{ atNtpAssociationEntry 11 }	The lowest measure of error associated with peer offset, based on delay, in seconds.
atNtpStatus	{ atNtp 11 }	Group of objects containing system status information. The objects in this group are not supported.
atNtpSysClockSync	{ atNtpStatus 1 }	Not supported.
atNtpSysStratum	{ atNtpStatus 2 }	Not supported.
atNtpSysReference	{ atNtpStatus 3 }	Not supported.
atNtpSysFrequency	{ atNtpStatus 4 }	Not supported.
atNtpSysPrecision	{ atNtpStatus 5 }	Not supported.
atNtpSysRefTime	{ atNtpStatus 6 }	Not supported.
atNtpSysClkOffset	{ atNtpStatus 7 }	Not supported.
atNtpSysRootDelay	{ atNtpStatus 8 }	Not supported.
atNtpSysRootDisp	{ atNtpStatus 9 }	Not supported.

AT-PRODUCTS-MIB

AT-PRODUCT-MIB defines object identifiers for Allied Telesis products. Objects in this MIB have the object identifier products ({ alliedTelesis 1 }) OID 1.3.6.1.4.1.207.1.

Table 95-16 lists object identifiers for products supported by the AlliedWare Plus™ Operating System.

Table 95-16: Object identifiers for Allied Telesis products supported by the AlliedWare Plus™ Operating System

Object	Object Identifier	Description
products	{ alliedTelesis 1 }	
swhub	{ products 4 }	Subtree beneath which switching hubs are defined.
at_x200_GE52T	{ swhub 181 }	x200-GE52T layer two switch
at_x200_GE28T	{ swhub 182}	x200-GE28T layer two switch
at_x210_9GT	{ swhub 196}	x210-9GT, 8xGigbit, 1xSFP/T
at_x210_16GT	{ swhub 197}	x210-16GT, 14xGigbit, 2xcombo SFP/T
at_x210_24GT	{ swhub 198}	x210-24GT, 20xGigbit, 4xcombo SFP/T
at_x310_26FT	{ swhub 216}	AT-x310-26FT layer two switch with 24 10/100 Base-T ports, two 10/100/1000 Base-T ports, and four 10 Gb/s SFP+ ports.
at_x310_50FT	{ swhub 217}	AT-x310-50FT layer two switch with 48 10/100 Base-T ports, two 10/100/1000 Base-T ports, and four 10 Gb/s SFP+ ports.
at_x310_26FP	{ swhub 218}	AT-x310-26FP layer two switch with 24 10/100 Base-T ports, two 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports, plus PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
at_x310_50FP	{ swhub 219}	AT-x310-50FP layer two switch with 48 10/100 Base-T ports, two 10/100/1000 Base-T ports, four 10 Gb/s SFP+ ports, plus PSE function available on pins 1/2 and 3/6 (Mode A) of every copper port.
routerSwitch	{ products 14 }	Subtree beneath which router and (non industrial) switch product MIB object IDs are assigned.
at_SwitchBladex908	{ routerSwitch 69 }	Switchblade x908 8 Slot Layer 3 Switch Chassis
at_x900_12XTS	{ routerSwitch 70 }	AT-x900-12XT/S Advanced Gigabit Layer 3+ Expandable Switch, 12 x combo ports (10/100/1000BASE-T copper or SFP), 1 x 30Gbps expansion bay
at_x900_24XT	{ routerSwitch 75 }	x900-24XT Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays
at_x900_24XS	{ routerSwitch 76 }	x900-24XS Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays
at_x900_24XT_N	{ routerSwitch 77 }	x900-24XT-N Enhanced Gigabit Layer 3+ Expandable Switch, 24 x 10/100/1000BASE-T copper ports (RJ-45 connectors), 2 x 20 Gigabit expansion bays, NEBS compliant
at_x600_24Ts	{ routerSwitch 80 }	x600-24Ts Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports
at_x600_24TsXP	{ routerSwitch 81 }	x600-24Ts/XP Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports, 2 x XFP ports
at_x600_48Ts	{ routerSwitch 82 }	x600-48Ts Stackable Managed L2+/L3 Ethernet Switch, 48 x 1000BASE-T copper ports, 4 x SFP ports

Table 95-16: Object identifiers for Allied Telesis products supported by the AlliedWare Plus™ Operating

Object	Object Identifier	Description
at_x600_48TsXP	{ routerSwitch 83 }	x600-48Ts/XP Stackable Managed L2+/L3 Ethernet Switch, 48 x 1000BASE-T copper ports, 4 x SFP ports, 2 x XFP ports
at_x600-24TsPoE	{ routerSwitch 91}	x600-24Ts-POE Stackable Managed L2+/L3 Ethernet PoE Switch, 24 x 1000BASE-T PoE ports, 4 x SFP (combo) ports
at_x600_24TPoEPlus	{routerSwitch 92}	x600-24Ts-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 24 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports
x610_48Ts_X_POEPlus	{routerSwitch 93}	x610-48Ts/X-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 48 x 1000BASE-T PoE+ ports, 2 x SFP (combo) ports, 2 x SFP+ ports
x610_48Ts_POEPlus	{routerSwitch 94}	x610-48Ts-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 48 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports
x610_24Ts_X_POEPlus	{routerSwitch 95}	x610-24Ts/X-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 24 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports, 2 x SFP+ ports
x610_24Ts_POEPlus	{routerSwitch 96}	x610-24Ts-POE+ Stackable Managed L2+/L3 Ethernet PoE+ Switch, 24 x 1000BASE-T PoE+ ports, 4 x SFP (combo) ports
x610_48Ts_X	{routerSwitch 97}	x610-48Ts/X Stackable Managed L2+/L3 Ethernet Switch, 48 x 1000BASE-T copper ports, 2 x SFP (combo) ports, 2 x SFP+ ports
x610_48Ts	{routerSwitch 98}	x610-48Ts Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports
x610_24Ts_X	{routerSwitch 99}	x610-24Ts/X Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports, 2 x SFP+ ports
x610_24Ts	{routerSwitch 100}	x610-24Ts Stackable Managed L2+/L3 Ethernet Switch, 24 x 1000BASE-T copper ports, 4 x SFP (combo) ports
x610_24SP_X	{routerSwitch 101}	x610-24SP/X Stackable Managed L2+/L3 Ethernet Switch, 24 x SFP (combo) ports, 2 x SFP+ ports
x510_28GTX	{routerSwitch 109}	x510-28GTX Stackable Managed L2+/L3 Ethernet Switch with 24 x 10/100/1000 Base-T ports and 4 x 10 Gb/s SFP+ ports.
x510_28GPX	{routerSwitch 110}	x510-28GPX Stackable Managed L2+/L3 Ethernet Switch with 24 x 10/100/1000 Base-T ports with PoE, 4 x 10 Gb/s SFP+ ports.
x510_28GSX	{routerSwitch 111}	x510-28GSX Stackable Managed L2+/L3 Ethernet Switch with 24 x 100/1000 SFP ports and 4 x 10 Gb/s SFP+ ports.
x510_52GTX	{routerSwitch 112}	x510-52GTX Stackable Managed L2+/L3 Ethernet Switch with 48 x 10/100/1000 Base-T ports and 4 x 10 Gb/s SFP+ ports.
x510_52GPX	{routerSwitch 113}	x510-52GPX Stackable Managed L2+/L3 Ethernet Switch with 48 x 10/100/1000 Base-T ports with PoE, and 4 x 10 Gb/s SFP+ ports.
at-SBx8106	{routerSwitch 114}	AT-SBx8106, SwitchBlade x8106 chassis.
x510DP_52GTX	{routerSwitch 116}	x510DP-52GTX Stackable Managed L2+/L3 Ethernet Switch with 48 x 10/100/1000 Base-T ports and 4 x 10 Gb/s SFP+ ports.

Table 95-16: Object identifiers for Allied Telesis products supported by the AlliedWare Plus™ Operating

Object	Object Identifier	Description
IX5_28GPX	{routerSwitch 117}	IX5-28GPX Stackable Managed L2+ Ethernet Switch with 24 x 10/100/1000 Base-T ports with PoE, 4 x 10 Gb/s SFP+ ports.
industrialSwitch	{ products 24 }	Subtree beneath which industrial switch product MIB object IDs are assigned.
at_IE200_6GT	{industrialSwitch 1}	IE200-6GT L2+ managed industrial Switch with 4 x 10/100/1000T LAN ports and 2 x SFP uplinks (100/1000X).
at_IE200_6GP	{industrialSwitch 2}	IE200-6GP L2+ managed industrial Switch with 4 x 10/100/1000T LAN ports (with 802.3at PoE+) and 2 x SFP uplinks (100/1000X).
at_IE200_6GPW	{industrialSwitch 3}	IE200-6GPW L2+ managed industrial Switch with 4 x 10/100/1000T LAN ports (with 802.3at PoE+) and 2 x SFP uplinks (100/1000X) and 802.11bgn wireless.

AT-RESOURCE-MIB

The AT-RESOURCE-MIB contains objects for displaying system hardware resource and host information (**Table 95-17**). Objects in this group have the object identifier rsc ({ sysinfo 21 }).

Table 95-17: Objects defined in AT-RESOURCE-MIB

Object and OID	Description
resource { sysinfo 21 }	Contains objects for displaying system hardware resource and host information.
rscBoardTable { resource 1}	Table containing information about boards installed in a device. Indexed by: 1. rscStkId 2. rscResourceId
rscBoardEntry { rscBoardTable 1 }	Information about a single board installed in the device.
rscStkId { rscBoardEntry 1 }	The ID of the stack member. It is a number from 1 to 8, assigned to a stackable unit by the operating system when it is stacked. A default of 1 is given to a stand-alone unit.
rscResourceId { rscBoardEntry 2 }	The resource ID number of the board. It is a number assigned to a hardware resource when the operating system detects its existence. Can be a value in range 1 to 4294967294.
rscBoardType { rscBoardEntry 3 }	The type of board. Can be one of the following: 1. Base 2. Expansion 3. Fan module 4. PSU
rscBoardName { rscBoardEntry 4 }	The name of the board. Can be one of the following: 1. SwitchBlade x908 2. XEM-12S 3. AT-PWR05-AC
rscBoardId { rscBoardEntry 5 }	The ID number of the board. Its value is an Allied Telesis assigned number, such as 274 for the XEM-12S, or 255 for the AT-9924Ts.
rscBoardBay { rscBoardEntry 6 }	The board installation location. Its value can be Bay1, Bay2, PSU1. For a base board, it has a value of a single character space.
rscBoardRevision { rscBoardEntry 7 }	The revision number of the board.
rscBoardSerialNumber { rscBoardEntry 8 }	The serial number of the board.
hostInfoTable { resource 2 }	Table containing general system information. Indexed by rscStkId.
hostInfoEntry { hostInfoTable 1 }	Information about a single system parameter
hostInfoDRAM { hostInfoTable 2 }	The host DRAM information.
hostInfoFlash { hostInfoTable 3 }	The host Flash information.
hostInfoUptime { hostInfoTable 4 }	The host up-time.
hostInfoBootloaderVersion { hostInfoTable 5 }	The host boot loader version.

AT-SETUP-MIB

AT-SETUP-MIB defines objects for managing software installation and configuration files (**Figure 95-6**, **Table 95-18**). Objects in this group have the object identifier setup ({ modules 500 }). The procedure in **Table 93-6 on page 93.22** shows how to use these MIB objects to upgrade to a new software version and boot configuration file. For objects used for file copying, see **"AT-FILEv2-MIB" on page 95.31**.

Figure 95-6: The AT-SETUP-MIB sub-tree

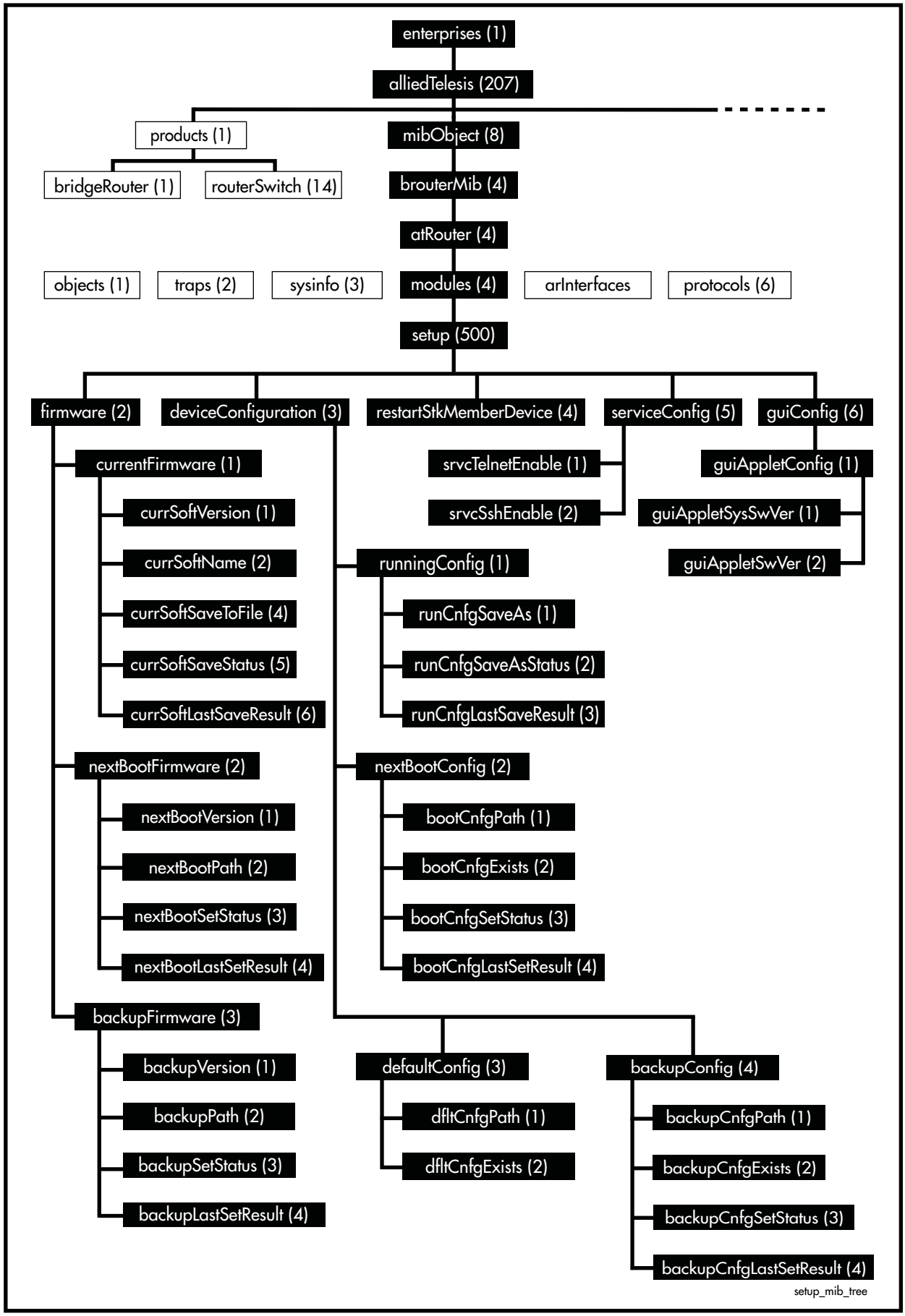


Table 95-18: Objects defined in AT-SETUP-MIB

Object Identifier	Description	Object Identifier
restartDevice	{ setup 1 }	Object for restarting the device. When set to '1', the device will restart immediately. Note: This object has been deprecated. Use instead the restartStkMemberDevice object.
firmware	{ setup 2 }	Objects for managing the software version files that the device will install and run.
currentFirmware	{ firmware 1 }	Information about the current software version installed on the device.
currSoftVersion	{ currentFirmware 1 }	Current software version.
currSoftName	{ currentFirmware 2 }	Current software name.
currSoftSaveAs	{ currentFirmware 3 }	The file name to save the currently running software to the root of the Flash. Only one save operation can be executed at a time across all SNMP users. Note: This object has been deprecated. Use instead the currSoftSaveToFile, currSoftSaveStatus and currSoftLastSaveResult objects.
currSoftSaveToFile	{ currentFirmware 4 }	Set with a URL to save the currently running software to the root of Flash or USB flash drive (e.g. 'flash:/filename.rel' or 'USB:/filename.rel'). The URL must not contain whitespace characters. Only one save operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of currSoftSaveStatus is 'idle'. Immediately upon executing the set action, the actual firmware save operation is started and will continue on the device until it has completed or a failure occurs. When read, this object will return the URL of the last firmware save operation that was attempted.
currSoftSaveStatus	{ currentFirmware 5 }	This object will return the status of any current operation to store the running software to a release file. The following values may be returned: 1. (idle) - there is no release file save operation in progress 2. (success) - the last release file save operation completed successfully 3. (failure) - the last release file save operation failed 4. (saving) - a release file save operation is currently in progress When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading currSoftLastSaveResult.
currSoftLastSaveResult	{ currentFirmware 6 }	Gives an indication of the result of the last completed SNMP operation to save the running firmware to a release file.

Table 95-18: Objects defined in AT-SETUP-MIB

Object Identifier	Description	Object Identifier
nextBootFirmware	{ firmware 2 }	Information about the software version to be installed on the device when booting.
nextBootVersion	{ nextBootFirmware 1 }	Provides information on the software version (major.minor.interim, for example version 5.4.1) that the device will boot from. A zero will be returned if the version cannot be determined.
nextBootPath	{ nextBootFirmware 2 }	The full path to the release file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of nextBootSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current boot firmware. Otherwise, the path should be of the form 'flash:/filename.cfg' or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must not be the same as the backup release file ■ it must have a .rel suffix ■ it must pass several internal checks to ensure that it is a genuine release file ■ in a stacked environment, there must be enough disk space available to store the release file on each stack member
nextBootSetStatus	{ nextBootFirmware 3 }	Returns the status of any current operation to set the next boot release file. The following values may be returned: ■ 1 (idle) - there is no boot release setting operation in progress ■ 2 (success) - the last boot release setting operation completed successfully ■ 3 (failure) - the last boot release setting operation failed ■ 5 (syncing) - a boot release setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading nextBootLastSetResult.
nextBootLastSetResult	{ nextBootFirmware 4 }	Gives an indication of the result of the last completed SNMP operation to set the boot release filename.

Table 95-18: Objects defined in AT-SETUP-MIB

Object Identifier	Description	Object Identifier
backupFirmware	{ firmware 3 }	Information about the backup software version and path.
backupVersion	{ backupFirmware 1 }	Provides information on the backup software version (major.minor.interim, for example version 5.4.1) that the device will boot from. A zero will be returned if the version cannot be determined.
backupPath	{ backupFirmware 2 }	The full path to the backup release file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of backupSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current backup firmware. Otherwise, the path should be of the form 'flash:/filename.cfg' or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must not be the same as the configured main release file ■ it must have a .rel suffix ■ it must pass several internal checks to ensure that it is a genuine release file ■ in a stacked environment, there must be enough disk space available to store the release file on each stack member
backupSetStatus	{ backupFirmware 3 }	Returns the status of any current operation to set the backup boot release file. The following values may be returned: ■ 1 (idle) - there is no backup boot release setting operation in progress ■ 2 (success) - the last backup boot release setting operation completed successfully ■ 3 (failure) - the last backup boot release setting operation failed ■ 5 (syncing) - a backup boot release setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading backupLastSetResult.
backupLastSetResult	{ backupFirmware 4 }	Gives an indication of the result of the last completed SNMP operation to set the backup boot release filename.
deviceConfiguration	{ setup 3	Objects for managing device configuration.
runningConfig	{ deviceConfiguration 1 }	

Table 95-18: Objects defined in AT-SETUP-MIB

Object Identifier	Description	Object Identifier
runCnfgSaveAs	{ runningConfig 1 }	Set with a URL to save the currently running software to the root of Flash or USB flash drive (e.g. 'flash:/filename.rel' or 'usb:/filename.rel'). The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of runCnfgSaveAsStatus is 'idle'. Immediately upon executing the set action, the system will attempt to save the running configuration and the process will continue on the device until it has completed or a failure occurs. When read, this object will return the URL of the last firmware save operation that was attempted.
runCnfgSaveAsStatus	{ runningConfig 2 }	Returns the status of any current operation to save the running configuration. The following values may be returned: 1. (idle) - there is no config file save operation in progress 2. (success) - the last config file save operation completed successfully 3. (failure) - the last config file save operation failed 4. (saving) - a config file save operation is currently in progress When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading runCnfgLastSaveResult.
runCnfgLastSaveResult	{ runningConfig 3 }	Gives an indication of the result of the last completed SNMP operation to save the running configuration.
nextBootConfig	{ deviceConfiguration 2 }	
bootCnfgPath	{ nextBootConfig 1 }	The full path to the configuration file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of bootCnfgSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current boot configuration. Otherwise, the path should be of the form 'flash:/myconfig.cfg' or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must have a .cfg suffix ■ in a stacked environment, there must be enough disk space available to store the configuration file on each stack member

Table 95-18: Objects defined in AT-SETUP-MIB

Object Identifier	Description	Object Identifier
bootCnfgExists	{ nextBootConfig 2 }	This object will return the value TRUE if the currently defined boot configuration file exists, or FALSE if it does not.
bootCnfgSetStatus	{ nextBootConfig 3 }	Returns the status of any current operation to set the next boot configuration file. The following values may be returned: ■ 1 (idle) - there is no boot configuration setting operation in progress ■ 2 (success) - the last boot configuration setting operation completed successfully ■ 3 (failure) - the last boot configuration setting operation failed ■ 5 (syncing) - a boot configuration setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading bootCnfgLastSetResult.
bootCnfgLastSetResult	{ nextBootConfig 4 }	Gives an indication of the result of the last completed SNMP operation to set the boot configuration filename.
defaultConfig	{ deviceConfiguration 3 }	
dfltCnfgPath	{ defaultConfig 1 }	The full path of the configuration file to use as backup when the device is rebooted. This object is not settable. The default configuration file is always 'flash:/default.cfg'.
dfltCnfgExists	{ defaultConfig 2 }	This object will return the value TRUE if the currently defined default configuration file exists, or FALSE if it does not.
backupConfig	{ deviceConfiguration 4 }	
backupCnfgPath	{ backupConfig 1 }	The full path to the backup configuration file that will be used the next time the device is rebooted. The URL must not contain whitespace characters. Only one set operation can be executed at a time across all SNMP users and an operation may not be started unless the current value of backupCnfgSetStatus is 'idle'. Immediately upon executing the set action, the system will attempt to set the new backup configuration path, and the process will continue on the device until it has completed or a failure occurs. This object can be set with an empty string in order to clear the current boot configuration. Otherwise, the path should be of the form 'flash:/myconfig.cfg' or 'card:/filename.cfg'. In order to set this object, the file must meet the following conditions: ■ it must exist ■ it must be located in the root of Flash (on the active master in a stacked environment) or USB flash drive ■ it must have a .cfg suffix ■ in a stacked environment, there must be enough disk space available to store the configuration file on each stack member

Table 95-18: Objects defined in AT-SETUP-MIB

Object Identifier	Description	Object Identifier
backupCnfgExists	{ backupConfig 2 }	This object will return the value TRUE if the currently defined backup configuration file exists, or FALSE if it does not.
backupCnfgSetStatus	{ backupConfig 3 }	Returns the status of any current operation to set the next backup boot configuration file. The following values may be returned: ■ 1 (idle) - there is no backup boot configuration setting operation in progress ■ 2 (success) - the last backup boot configuration setting operation completed successfully ■ 3 (failure) - the last backup boot configuration setting operation failed ■ 5 (syncing) - a backup boot configuration setting operation is currently in progress and the file is being synchronized across the stack When a read of this object returns a value of 'success' or 'failure', it will immediately be reset to 'idle' and a new operation may be initiated if desired. A detailed description of the last completed operation may be determined by reading backupCnfgLastSetResult.
backupCnfgLastSetResult	{ backupConfig 4 }	Gives an indication of the result of the last completed SNMP operation to set the backup boot configuration filename.
restartStkMemberDevice	{ setup 4 }	This object causes a specified device to restart immediately. The restart is initiated by setting its value to the device's stack member ID. Setting its value to zero will cause all devices in the stack, or a standalone device, to restart. Reading the object will always return zero.
serviceConfig	{ setup 5 }	
srvcTelnetEnable	{ serviceConfig 1 }	This object is used to either read or set the state of the telnet server on a device. Telnet can be enabled by setting the value of this object to ‘enable(1)’ or can be disabled by setting the value ‘disable(2)’.
srvcSshEnable	{ serviceConfig 2 }	This object is used to either read or set the state of the SSH server on a device. SSH can be enabled by setting the value of this object to ‘enable(1)’ or can be disabled by setting the value ‘disable(2)’.
guiConfig	{ setup 6 }	
guiAppletConfig	{ guiConfig 1 }	
guiAppletSysSwVer	{ guiAppletConfig 1 }	This object represents the system software release that the currently selected GUI applet was designed to run on. The system automatically searches for GUI applet files that reside in the root directory of the Flash memory, and selects the latest available file that is applicable to the currently running system software. This is the applet that will be uploaded to a user's web browser when they initiate the GUI.
	{ guiAppletConfig 2 }	This object represents the software version of the currently selected GUI applet. The system automatically searches for GUI applet files residing in the root directory of the Flash memory, and selects the latest available one that is applicable to the currently running system software. This is the applet that will be uploaded to a user's web browser when they initiate the GUI.

AT-SMI-MIB

AT-SMI-MIB defines the high-level structure and root objects of the Allied Telesis Enterprise MIB (**Table 95-19**). These objects are imported by other component MIBs of the Allied Telesis Enterprise MIB.

Table 95-19: AT Enterprise MIB - High Level Structure

Object	Object Identifier	Description
alliedTelesis	{ enterprises 207 } 1.3.6.1.4.1.207	Root of the Allied Telesis Enterprise MIB under the private(4) node defined in RFC1155-SMI.
products	{ alliedTelesis 1 } 1.3.6.1.4.1.207.1	Sub-tree of all product OIDs. Described in **"AT-PRODUCTS-MIB" on page 95.51**.
bridgeRouter	{ products 1 } 1.3.6.1.4.1.207.1.1	Sub-tree of bridge product MIB objects (not applicable for AlliedWare Plus).
routerSwitch	{ products 14 } 1.3.6.1.4.1.207.1.2	Sub-tree for all router and switch product MIB objects.
industrialSwitch	{ products 24 } 1.3.6.1.4.1.207.1.24	Sub-tree for industrial switch product MIB objects.
mibObject	{ alliedTelesis 8 } 1.3.6.1.4.1.207.8	Sub-tree for all managed objects.
brouterMib	{ mibObject 4 } 1.3.6.1.4.1.207.8.4	Sub-tree of objects for managing bridges, routers, and switches.
atRouter	{ brouterMib 4 } 1.3.6.1.4.1.207.8.4.4	Sub-tree of objects for managing multiprotocol routers and switches.
objects	{ atRouter 1 } 1.3.6.1.4.1.207.8.4.4.1	Sub-tree of OIDs for boards, releases, interface types, and chips.
traps	{ atRouter 2 } 1.3.6.1.4.1.207.8.4.4.2	Sub-tree for generic traps (not applicable for AlliedWare Plus).
sysinfo	{ atRouter 3 } 1.3.6.1.4.1.207.8.4.4.3	Sub-tree of objects describing general system information.
modules	{ atRouter 4 } 1.3.6.1.4.1.207.8.4.4.4	Sub-tree of objects for monitoring and managing software features.
arInterfaces	{ atRouter 5 } 1.3.6.1.4.1.207.8.4.4.5	Sub-tree of objects describing boards, slots and physical interfaces.
protocols	{ atRouter 6 } 1.3.6.1.4.1.207.8.4.4.6	Sub-tree of OIDs for protocols.
atAgents	{ atRouter 7 } 1.3.6.1.4.1.207.8.4.4.7	Sub-tree of objects describing variations from standards.

Table 95-20 lists the major modules of the AT-SMI-MIB grouped by their object identifiers. Note that this is also the order in which they are described in this chapter.

Table 95-20: AT-SMI-MIBs Listed by Object Group

MIB Section	OID	Description
AT-SMI-MIB		This section describes the structure of management information for the Allied Telesis Enterprise object, alliedTelesis { 1.3.6.1.4.1.207 }.
AT-PRODUCTS-MIB	1.3.6.1.4.1.207.1	Object identifiers for Allied Telesis products. See **"AT-PRODUCTS-MIB" on page 95.51**.

Table 95-20: AT-SMI-MIBs Listed by Object Group(cont.)

MIB Section	OID	Description
AT-BOARDS-MIB	1.3.6.1.4.1.207.8.4.4.1.1	Object identifiers for boards, interface types, and chip sets. See **"AT-BOARDS-MIB" on page 95.13**.
AT-SYSINFO-MIB	1.3.6.1.4.1.207.8.4.4.3	Objects that describe generic system information and environmental monitoring. See **"AT-SYSINFO-MIB" on page 95.65**.
AT-ENVMONv2-MIB	1.3.6.1.4.1.207.8.4.4.3.12	Objects and traps for monitoring fans, voltage rails, temperature sensors, and power supply bays. See **"AT-ENVMONv2-MIB" on page 95.21**.
AT-VCSTACK-MIB	1.3.6.1.4.1.207.8.4.4.3.13	Objects for managing Virtual Chassis Stacking (VCS). See **"AT-ENVMONv2-MIB" on page 95.21**.
AT-MIBVERSION-MIB	1.3.6.1.4.1.207.8.4.4.3.15	Object to display the last software release that contained changes to the support AT Enterprise MIB definition files. See **"AT-MIBVERSION-MIB" on page 95.47**.
AT-USER-MIB	1.3.6.1.4.1.207.8.4.4.3.20	Objects for displaying information of users currently logged into a device, or configured in the Local User Data base of the device. See **"AT-USER-MIB" on page 95.71**.
AT-RESOURCE-MIB	1.3.6.1.4.1.207.8.4.4.3.21	Objects for displaying system hardware resource information. See **"AT-RESOURCE-MIB" on page 95.54**.
AT-LICENSE-MIB	1.3.6.1.4.1.207.8.4.4.3.22	Objects for managing software licenses on devices using AlliedWare Plus™ Operating System. See **"AT-LICENSE-MIB" on page 95.40**.
AT-TRIGGER-MIB	1.3.6.1.4.1.207.8.4.4.4.53	Objects for managing triggers. See **"AT-TRIGGER-MIB" on page 95.69**.
AT-LOOPPROTECT-MIB	1.3.6.1.4.1.207.8.4.4.4.54	Objects for managing Allied Telesis Loop Protection. See **"AT-LOOPPROTECT-MIB" on page 95.45**.
AT-SETUP-MIB	1.3.6.1.4.1.207.8.4.4.4.500	Objects for managing software installation and configuration files. See **"AT-SETUP-MIB" on page 95.55**.
AT-DNS-CLIENT-MIB	1.3.6.1.4.1.207.8.4.4.4.501	Objects for managing Allied Telesis DNS Client Configuration. See **"AT-DNS-CLIENT-MIB" on page 95.20**.
AT-NTP-MIB	1.3.6.1.4.1.207.8.4.4.4.502	Objects for managing Allied Telesis Network Time Protocol (NTP) configuration. See **"AT-NTP-MIB" on page 95.48**.
AT-EPSRv2-MIB	1.3.6.1.4.1.207.8.4.4.4.536	Objects for managing Allied Telesis EPSR. See **"AT-EPSRv2-MIB" on page 95.29**.
AT-DHCPSN-MIB	1.3.6.1.4.1.207.8.4.4.4.537	Objects for managing Allied Telesis DHCP Snooping. See **"AT-DHCPSN-MIB" on page 95.17**.
AT-FILEv2-MIB	1.3.6.1.4.1.207.8.4.4.4.600	Objects for displaying and managing file content on local, stacked, and remote sources. See **"AT-FILEv2-MIB" on page 95.31**.
AT-LOG-MIB	1.3.6.1.4.1.207.8.4.4.4.601	Objects for listing log entries from the buffered and permanent logs. See **"AT-LOG-MIB" on page 95.43**.
AT-IP-MIB	1.3.6.1.4.1.207.8.4.4.4.602	Objects for Allied Telesis specific IP address management. See **"AT-IP-MIB" on page 95.38**.
AT-ALMMON-MIB	1.3.6.1.4.1.207.8.4.4.3.26	Objects for managing Allied Telesis Alarm Monitor.

AT-SYSINFO-MIB

AT-SYSINFO-MIB defines objects that describe generic system information and environmental monitoring. Objects in this group have the object identifier sysinfo ({ atRouter 3 }). **Table 95-21** lists the objects supported by the AlliedWare Plus™ sysinfo MIB.

Table 95-21: Objects defined in AT-SYSINFO-MIB

Object	Description
sysinfo { atRouter 3 } (1.3.6.1.4.1.207.8.4.4.3)	Subtree containing generic system information.
fanAndPs {sysinfo 1 } (1.3.6.1.4.1.207.8.4.4.3.1)	A collection of objects for monitoring fans and power supplies. For devices running the AlliedWare Plus™ Operating System, these objects are superceded by objects in the AT-ENVMON-MIB (see **"AT-ENVMONv2-MIB" on page 95.21**).
restartGroup {sysinfo 2 }	A collection of objects and traps for activating and monitoring restarts. This group is not supported by devices running the AlliedWare Plus™ Operating System.
cpu {sysinfo 3 }	A collection of objects containing information about the CPU utilization over different periods of time. All values are expressed as a percentage - integer in range 0 to 100.
cpuUtilisationMax {cpu 1 }	Maximum CPU utilization since the device was last restarted.
cpuUtilisationAvg {cpu 2 }	Average CPU utilization since the device was last restarted.
cpuUtilisationAvgLastMinute {cpu 3 }	Average CPU utilization over the past minute.
cpuUtilisationAvgLast10Seconds {cpu 4 }	Average CPU utilization over the past ten seconds.
cpuUtilisationAvgLastSecond {cpu 5 }	Average CPU utilization over the past second.
cpuUtilisationAvgMaxLast5Minutes {cpu 6 }	Maximum CPU utilization over the last 5 minutes.
cpuUtilisationAvgLast5Minutes {cpu 7 }	Average CPU utilization over the past 5 minutes.
cpuUtilisationStackTable {cpu 8 }	A list of stack members.
cpuUtilisationStackEntry {cpuUtilisationStackTable 1}	A set of parameters that describe the CPU utilization of a stack member
cpuUtilisationStackId {cpuUtilisationStackEntry 1}	Stack member ID.
cpuUtilisationStackMax {cpuUtilisationStackEntry 2}	Maximum CPU utilization since the router was last restarted. Expressed as a percentage.
cpuUtilisationStackAvg {cpuUtilisationStackEntry 3}	Average CPU utilization since the router was last restarted. Expressed as a percentage.
cpuUtilisationStackAvgLastMinute {cpuUtilisationStackEntry 4}	Average CPU utilisation over the past minute. Expressed as a percentage.

Table 95-21: Objects defined in AT-SYSINFO-MIB

Object	Description
cpuUtilisationStackAvgLast10Seconds {cpuUtilisationStackEntry 5}	Average CPU utilisation over the past ten seconds. Expressed as a percentage.
cpuUtilisationStackAvgLastSecond {cpuUtilisationStackEntry 6}	Average CPU utilisation over the past second. Expressed as a percentage.
cpuUtilisationStackMaxLast5Minutes {cpuUtilisationStackEntry 7}	Maximum CPU utilisation over the last 5 minutes. Expressed as a percentage.
cpuUtilisationStackAvgLast5Minutes {cpuUtilisationStackEntry 8}	Average CPU utilisation over the past 5 minutes. Expressed as a percentage.
sysTemperature {sysinfo 4 }	A collection of objects and traps for monitoring and managing the temperature status. For devices running the AlliedWare Plus™ Operating System.
atContactDetails {sysinfo 5 }	Contact details for Allied Telesis.
memory {sysinfo 7 }	A collection of objects and traps for monitoring memory usage and status.
atEnvMonv2 {sysinfo 12 }	AT Environment Monitoring v2 MIB for managing and reporting data relating to voltage rails, fan speeds, temperature sensors and power supply units. Objects under this portion of the OID are shown in the **"AT-ENVMONv2-MIB" on page 95.21**.
vcstack {sysinfo 13 }	A collection of objects for managing Virtual Chassis Stacking in AlliedWare Plus™. See **"AT-VCSTACK-MIB" on page 95.73**.
atPortInfo {sysinfo 14 }	Objects containing information about the transceiver of an interface. This portion of the object tree is documented separately in: **"AT-PORTINFO" on page 95.67**.
atVlanInfo {sysinfo 16 }	A collection of objects for counting bytes or incoming frames within a selected VLAN. Note that these objects are only appropriate for the IX5, x510 and x610 series products. Objects under this portion of the OID are shown in the **"AT-VLANINFO-MIB" on page 95.78**.
{sysinfo 17 } to {sysinfo 19 }	These objects are not supported on your switch.
user {sysinfo 20 }	Contains objects for displaying information of users currently logged into a device, or configured in its local database. Objects under this portion of the OID are shown in the **"AT-USER-MIB" on page 95.71**.
resource {sysinfo 21 }	Contains objects for displaying hardware resource information. Objects under this portion of the OID are shown in the **"AT-RESOURCE-MIB" on page 95.54**.
license {sysinfo 22 }	This MIB, is used for listing applied software licenses, adding new licenses, and deleting existing licenses. Objects under this portion of the OID are shown in the **"AT-LICENSE-MIB" on page 95.40**.
chassis {sysinfo 23 }	This MIB is used for accessing trap notifications on chassis based products. Note that these objects are only appropriate for the x8100 series products.

AT-PORTINFO

This table defines objects for managing interface port objects such as transceivers. Objects in this group have the object identifier atPortInfo ({ sysinfo 14 }), OID path, 1.3.6.1.4.1.207.8.4.4.3.14.

Table 95-22: Objects defined in AT-ATPORTINFO portion of the MIB

Object / Object Identifier	Description
atPortInfo {sysinfo 14}	This object returns information about interface transceivers.
atPortInfoTransceiverTable {atPortInfo 1}	A table of information about the transceiver of a interface.
atPortInfoTransceiverEntry {atPortInfoTransceiverTable 1}	The description, the transceiver type of a interface.
atPortInfoTransceiverifIndex {atPortInfoTransceiverEntry 1}	The ifIndex for the interface represented by this entry of the interfaces table.
atPortInfoTransceiverType {atPortInfoTransceiverEntry 2}	This object indicates the type of transceiver on a interface. It contains the following value list objects:
	The type of transceiver on an interface can be one of the following: ■ rj45(1) ■ sfp-px(2) ■ sfp-bx10(3) ■ sfp-fx(4) ■ sfp-100base-lx(5) ■ sfp-t(6) ■ sfp-cx(7) ■ sfp-zx-cwdm(8) ■ sfp-lx(9) ■ sfp-sx(10) ■ sfp-oc3-lr(11) ■ sfp-oc3-ir(12) ■ sfp-oc3-mm(13) ■ xfp-srsw(14) ■ xfp-lrlw(15) ■ xfp-erew(16) ■ xfp-sr(17) ■ xfp-lr(18) ■ xfp-er(19) ■ xfp-lrm(20) ■ xfp-sw(21) ■ xfp-lw(22) ■ xfp-ew(23)

Table 95-22: Objects defined in AT-ATPORTINFO portion of the MIB(cont.)

Object / Object Identifier	Description
	■ Tunknown(24) ■ empty(25) ■ sfpp-sr(26) ■ sfpp-lr(27) ■ sfpp-er(28) ■ sfpp-lrm(29) ■ inf-1-x-copper-pasv(30) ■ inf-1-x-copper-actv(31) ■ inf-1-x-lx(32) ■ inf-1-x-sx(33) ■ cx4(34)
atPortRenumberEvents {atPortInfo 2}	The number of times that port number values (represented by the dot1dBasePort object in BRIDGE-MIB), have been re-assigned due to stack member leave/join events or XEM hot-swap events, since the system was initialized.

AT-TRIGGER-MIB

AT-TRIGGER-MIB defines objects for managing triggers (**Table 95-23**). Objects in this group have the object identifier trigger ({ modules 53 }). All objects in this group have read only access.

Table 95-23: Objects defined in AT-TRIGGER-MIB

Object Identifier	Description
triggerTraps { trigger 0 }	Sub-tree for all trigger traps.
triggerTrap { triggerTraps 1 }	Notification generated when a trigger is activated. It returns the value of triggerLastTriggerActivated.
triggerLastTriggerActivated { trigger 1 }	Trigger number of the most recent trigger activated on the switch.
triggerConfigInfoTable { trigger 9 }	Table of information about each trigger that has been configured, indexed by triggerNumber.
triggerConfigInfoEntry { triggerConfigInfoTable 1 }	Information about the configuration of a single trigger.
triggerNumber { triggerConfigInfoEntry 1 }	ID number of the trigger. Values are in range 1- 250.
triggerName { triggerConfigInfoEntry 2 }	Name and description of the trigger.
triggerTypeDetail { triggerConfigInfoEntry 3 }	Trigger type and its activation conditions.
triggerActiveDaysOrDate { triggerConfigInfoEntry 4 }	The days of a week or the date on which the trigger can be activated.
triggerActivateAfter { triggerConfigInfoEntry 5 }	Time after which the trigger can be activated.
triggerActivateBefore { triggerConfigInfoEntry 6 }	Time before which the trigger can be activated.
triggerActiveStatus { triggerConfigInfoEntry 7 }	Whether or not the trigger can be activated.
triggerTestMode { triggerConfigInfoEntry 8 }	Whether or not the trigger is operating in diagnostic (test) mode.
triggerSnmpTrap { triggerConfigInfoEntry 9 }	Whether or a not an SNMP trap will be generated when the trigger is activated.
triggerRepeatTimes { triggerConfigInfoEntry 10 }	Whether the trigger can repeat an unlimited number of times (continuous) or a specified number of times. If the trigger can repeat only a specified number of times, then the number of times the trigger has already been activated is displayed in brackets.
triggerLasttimeModified { triggerConfigInfoEntry 11 }	Date and time that the trigger configuration was last modified.
triggerNumberOfActivation { triggerConfigInfoEntry 12 }	Number of times the trigger has been activated since the last restart of the device.
triggerLasttimeActivation { triggerConfigInfoEntry 13 }	Date and time that the trigger was last activated.
triggerNumberOfScripts { triggerConfigInfoEntry 14 }	Number of scripts that this trigger will execute. Values are in range 0-5.

Table 95-23: Objects defined in AT-TRIGGER-MIB(cont.)

Object Identifier	Description
triggerScript1 { triggerConfigInfoEntry 15 }	Name of the first script that this trigger will execute if the trigger is activated.
triggerScript2 { triggerConfigInfoEntry 16 }	Name of the second script that this trigger will execute if the trigger is activated.
triggerScript3 { triggerConfigInfoEntry 17 }	Name of the third script that this trigger will execute if the trigger is activated.
triggerScript4 { triggerConfigInfoEntry 18 }	Name of the fourth script that this trigger will execute if the trigger is activated.
triggerScript5 { triggerConfigInfoEntry 19 }	Name of the fifth script that this trigger will execute if the trigger is activated.
triggerCounters { trigger 10 }	Collection of counters for trigger activations.
triggerNumOfActivation { triggerCounters 1 }	Number of times a trigger has been activated.
triggerNumOfActivationToday { triggerCounters 2 }	Number of times a trigger has been activated today.
triggerNumOfPerodicActivationToday { triggerCounters 3 }	Number of times a periodic trigger has been activated today.
triggerNumOfInterfaceActivationToday { triggerCounters 4 }	Number of times an interface trigger has been activated today.
triggerNumOfResourceActivationToday { triggerCounters 5 }	Number of times a CPU or memory trigger has been activated today.
triggerNumOfRebootActivationToday { triggerCounters 6 }	Number of times a reboot trigger has been activated today.
triggerNumOfPingPollActivationToday { triggerCounters 7 }	Number of times a ping-poll trigger has been activated today.
triggerNumOfStackMasterFailActivationToday { triggerCounters 8 }	Number of times a stack master fail trigger has been activated today.
triggerNumOfStackMemberActivationToday { triggerCounters 9 }	Number of times a stack member trigger has been activated today.

AT-USER-MIB

The AT-USER-MIB contains objects for displaying information about users currently logged into a device, or configured in the Local User Database of the device (**Table 95-24**). Objects in this group have the object identifier user ({ sysinfo 20 }).

Table 95-24: Objects defined in AT-USER-MIB

Object	Object Identifier	Description
userInfoTable (207.8.4.4.3.20.1)	{ user 1 }	Table containing information about users. Each entry in the table represents a user currently logged into the device. Indexed by: rscBoardType and rscBoardIndex.
userInfoEntry	{ userInfoTable 1 }	Information about a single user logged into the device.
userInfoType	{ userInfoEntry 1 }	The type of connection through which the user logged into the device. Can be: 1. console (1) 2. aux (2) 3. telnet (3) 4. script (4) 5. stack (5)
userInfoIndex	{ userInfoEntry 2 }	Index of the line upon which the user logged into the device. Can be a value in range 1 to 16.
userInfoName	{ userInfoEntry 3 }	User name of the user logged into the device.
userInfoPrivilegeLevel	{ userInfoEntry 4 }	The user's privilege level. Can be a value in range 1 to 15.
userInfoIdleTime	{ userInfoEntry 5 }	The amount of time since the user was last active, in the form hh:mm:ss.
userInfoLocation	{ userInfoEntry 6 }	The user location or login method. It can be an IP Address used by the user to telnet into the device, or an asyn port.
userInfoPasswordLifetime	{ userInfoEntry 7 }	The number of days remaining until the user's password expires. Depending on the current user setting it will display one of the following: No Expiry - the password will never expire (default setting) x days - where x is the remaining lifetime of the current password (maximum lifetime value is 1000 days) -x days (expired) - indicating that the current password expired x days ago
userInfoPasswordLastChange	{ userInfoEntry 8 }	The number of days since the password was last altered.
userConfigTable	{ user 2 } or (207.8.4.4.3.20.2)	Table containing user configuration information. Each entry in the table relates to a user configured in the Local User Database of the device. Indexed by userConfigIndex.

Table 95-24: Objects defined in AT-USER-MIB(cont.)

Object	Object Identifier	Description
userConfigEntry	{ userConfigTable 1 }	Information about a single user configured in the Local User Database of the device.
userConfigIndex	{ userConfigEntry 1 }	Unique number used to identify entries in the userConfigTable.
userConfigName	{ userConfigEntry 2 }	The user's name.
userConfigPrivilegeLevel	{ userConfigEntry 3 }	The privilege level granted to the user. Can be a value in range 1 to 15.
userSecurityPasswordRules	{ user 3 } or (207.8.4.4.3.20.3)	Information about user password security rules.
userSecurityPasswordHistory	{ userSecurityPasswordRules 1 }	The number of previous passwords that are retained for comparison when a user password is created. A new password must be unique when compared against the previous history. A value of 0 represents no restriction. The maximum number of retained passwords is 15.
userSecurityPasswordLifetime	{ userSecurityPasswordRules 2 }	The maximum number of days that the password may persist before a change is required. A value of 0 represents no expiry. The maximum value is 1000.
userSecurityPasswordWarning	{ userSecurityPasswordRules 3 }	The number of days before the password expires that a warning message is displayed when the user logs in. A value of 0 indicates no warning. The maximum value is 1000 but must always be less than the password lifetime.
userSecurityPasswordMinLength	{ userSecurityPasswordRules 4 }	The minimum allowable password length.
userSecurityPasswordMinCategory	{ userSecurityPasswordRules 5 }	The minimum number of different categories that the password must satisfy to be considered valid. Categories are split into four groups: upper-case letters lower-case letters digits special symbols. ASCII characters not included in the previous three categories.
userSecurityPasswordForced	{ userSecurityPasswordRules 6 }	Whether or not a user with an expired password is forced to change their password at the next login. At login a user with an expired password is prompted to change their password. If the new password meets the current security password rules the user is allowed to log in, otherwise they are rejected.
userSecurityPasswordReject	{ userSecurityPasswordRules 7 }	Whether or not a user login attempt with an expired password is rejected. If the user is not rejected then they can log in.

AT-VCSTACK-MIB

AT-VCSTACK-MIB defines objects for managing Virtual Chassis Stacking (**Table 95-25**). Objects in this group have the object identifier vcstack ({ sysinfo 13 }).

Figure 95-7 on page 95.73 shows the tree structure of the AT-VCSTACK objects.

Figure 95-7: The AT-VCSTACK MIB sub-tree

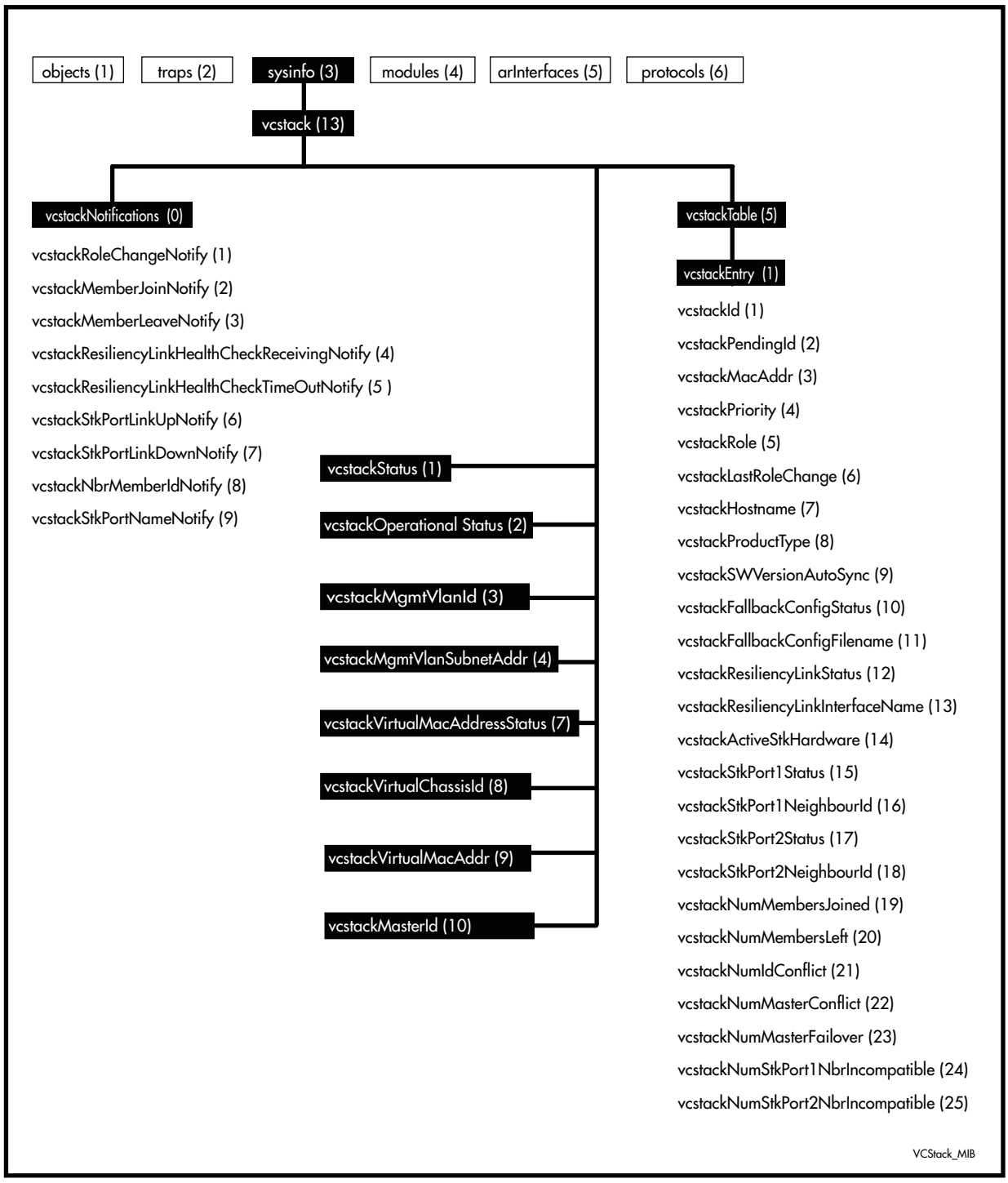


Table 95-25: Objects defined in AT-VCSTACK-MIB

Object	Object Identifier	Description
vcstack	{ sysinfo (13) }	Overall stack status.
vcstackNotifications	{ vcstack 0 }	List of traps (notifications) generated for the stack:
vcstackRoleChangeNotify	{ vcstackNotifications 1 }	The stack status can take one of the following states: 1. normalOperation (1) 2. operatingInFailoverState (2) 3. standaloneUnit (3) 4. ringTopologyBroken (4)
vcstackMemberJoinNotify	{ vcstackNotifications 2 }	Notification generated when a member joins the stack. Displays the objects: 1. vcstackId 2. vcstackNbrMemberId
vcstackMemberLeaveNotify	{ vcstackNotifications 3 }	Notification generated when a member leaves the stack. Displays the objects: 1. vcstackId 2. vcstackNbrMemberId
vcstackResiliencyLinkHealthCheckReceivingNotify	{ vcstackNotifications 4 }	Notification generated when the resiliency link is activated. Displays the objects: 1. vcstackId 2. vcstackResiliencyLinkInterfaceName
vcstackResiliencyLinkHealthCheckTimeOutNotify	{ vcstackNotifications 5 }	Notification generated when the backup member's receive timer has timed-out, indicating that the Backup has lost contact with the Master via the resiliency link. Displays the objects: 1. vcstackId 2. vcstackResiliencyLinkInterfaceName
vcstackStkPortLinkUpNotify	{ vcstackNotifications 6 }	Notification generated when the stack port link is up. Displays the objects: 1. vctasckId 2. vcstackStkPortName
vcstackStkPortLinkDownNotify	{ vcstackNotifications 7 }	Notification generated when the stack port link is down. Displays the objects: 1. vcstackId 2. vcstackStkPortName
vvcstackNbrMemberIdNotify	{ vcstackNotifications 8 }	The stack member id related to this trap
vcstackStkPortNameNotify	{ vcstackNotifications 9 }	The stack port name related to this trap

Table 95-25: Objects defined in AT-VCSTACK-MIB(cont.)

Object	Object Identifier	Description
vcstackOperationalStatus	{ vcstack 2 }	The operational status of the stack can be either: 1. enabled (1) 2. disabled (2)
vcstackMgmtVlanId	{ vcstack 3 }	The current stacking management VLAN ID
vcstackMgmtVlanSubnetAddr	{ vcstack 4 }	The current stacking management VLAN subnet address
vcstackTable	{ vcstack 5 }	Table of information about stack members, indexed by vcstackId.
vcstackEntry	{ vcstackTable 1 }	Information about a single stack member, indexed by vcstackId.
vcstackId	{ vcstackEntry 1 }	Stack member ID.
vcstackPendingId	{ vcstackEntry 2 }	Pending stack member ID.
vcstackMacAddr	{ vcstackEntry 3 }	Stack member's hardware MAC address.
vcstackPriority	{ vcstackEntry 4 }	Priority for election of the stack master. The lowest number has the highest priority.
vcstackRole	{ vcstackEntry 5 }	Stack member's role in the stack. Can be one of the following: 1. leaving (1) 2. discovering (2) 3. synchronizing (3) 4. backupMember (4) 5. pendingMaster (5) 6. disabledMaster (6) 7. fallbackMaster (7) 8. activeMaster (8)
vcstackLastRoleChange	{ vcstackEntry 6 }	Time and date when the stack member last changed its role in the stack.
vcstackHostname	{ vcstackEntry 7 }	Stack member's hostname.
vcstackProductType	{ vcstackEntry 8 }	Stack members product type.
vcstackSWVersionAutoSync	{ vcstackEntry 9 }	Whether or not the stack member's software is automatically upgraded.
vcstackFallbackConfigStatus	{ vcstackEntry 10 }	Status of the fallback configuration file. Can be one of: 1. fileExists (1) 2. fileNotFound (2) 3. notConfigured (3)

Table 95-25: Objects defined in AT-VCSTACK-MIB(cont.)

Object	Object Identifier	Description
vcstackFallbackConfigFilename	{ vcstackEntry 11 }	Filename of the fallback configuration file.
vcstackResiliencyLinkStatus	{ vcstackEntry 12 }	Status of the stack members resiliency link. Can be one of: 1. configured (1) 2. successful (2) 3. failed (3) 4. notConfigured (4)
vcstackResiliencyLinkInterfaceName	{ vcstackEntry 13 }	Name of the interface the resiliency link is configured on.
vcstackActiveStkHardware	{ vcstackEntry 14 }	Stack ports hardware type. Can be one of: 1. value (0) is now obsolete 2. xemStk (1) 3. builtinStackingPorts (2) 4. none (3) is now obsolete 5. stackXG (4)
vcstackStkPort1Status	{ vcstackEntry 15 }	Status of stack-port 1. Can be one of the following: 1. down (1) 2. neighbourIncompatible (2) 3. discoveringNeighbour (3) 4. learnedNeighbour (4)
vcstackStkPort1NeighbourId	{ vcstackEntry 16 }	ID of the neighbor on stack-port 1. Zero indicates no learned neighbor.
vcstackStkPort2Status	{ vcstackEntry 17 }	Status of stack-port 2. Can be one of: 1. down (1) 2. neighbourIncompatible (2) 3. discoveringNeighbour (3) 4. learnedNeighbour (4)
vcstackStkPort2NeighbourId	{ vcstackEntry 18 }	ID of the neighbor on stack-port 2. Zero indicates no learned neighbor.
vcstackNumMembersJoined	{ vcstackEntry 19 }	Number of times the stack has acquired a member.
vcstackNumMembersLeft	{ vcstackEntry 20 }	Number of times the stack has lost a member.
vcstackNumIdConflict	{ vcstackEntry 21 }	Number of times that a stack member ID conflict has occurred.
vcstackNumMasterConflict	{ vcstackEntry 22 }	Number of times that a stack master conflict has occurred.
vcstackNumMasterFailover	{ vcstackEntry 23 }	Number of times that the stack master has failed.

Table 95-25: Objects defined in AT-VCSTACK-MIB(cont.)

Object	Object Identifier	Description
vcstackNumStkPort1NbrIncompatible	{ vcstackEntry 24 }	Number of times that the neighbor on stack port 1 was incompatible.
vcstackNumStkPort2NbrIncompatible	{ vcstackEntry 25 }	Number of times that the neighbor on stack port 2 was incompatible.
vcstackVirtualMacAddressStatus	{vcstack 7}	Indicates whether the virtual MAC address is enabled or disabled. Read-only object.
vcstackVirtualChassisId	{vcstack 8}	Displays the current virtual chassis ID. Read-only object.
vcstackVirtualMacAddr	{vcstack 9}	Displays the virtual MAC address used by the stack. Read-only object.
vcstackMasterId	{vcstack 10}	Displays the stack ID of the master unit, or the stack ID of the standalone unit. Read-only object

AT-VLANINFO-MIB

The atVlanStatistics-MIB (**Figure 95-5**, and **Table 95-13**) defines objects for managing VLANs. The MIB contains a sub tree for managing VLAN statistics. Objects in the VLAN Statistics sub-tree have the object identifier ({atVlanInfo 1}).

Figure 95-8: The atVlanStatistics MIB tree

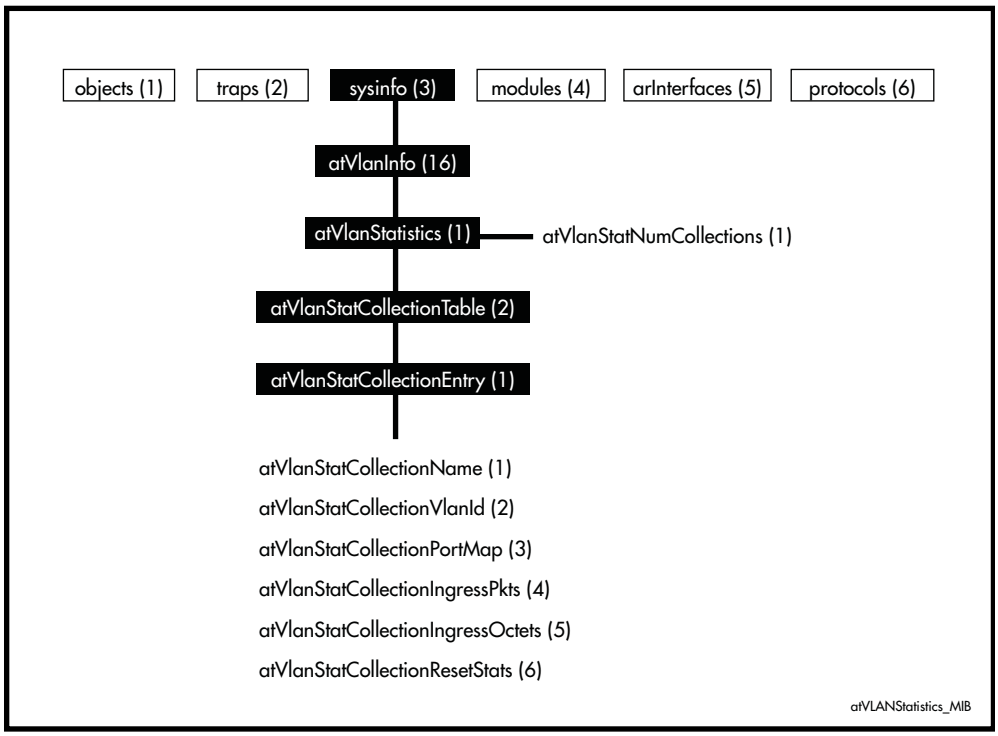


Table 95-26: AT Enterprise MIB - High Level Structure

Object	Description
vlaninfo {sysinfo 16}	Root of the Allied Telesis Enterprise MIB under the private(4) node defined in RFC1155-SMI.
atVlanStatistics {vlaninfo 1}	The number of unique VLAN statistic gathering instances defined on the device.
atVlanStatNumCollections {atVlanStatistics 1}	The number of unique VLAN statistic gathering instances defined on the device.
atVlanStatCollectionTable {atVlanStatistics 2}	A table of VLAN statistic instances.
atVlanStatCollectionEntry {atVlanStatCollectionTable 1}	Each entry represents a unique VLAN statistic gathering instance defined on the device. Sequences are:
atVlanStatCollectionName {atVlanStatCollectionEntry 1}	The name of a VLAN statistics collection instance.
atVlanStatCollectionVlanId {atVlanStatCollectionEntry 2}	The VLAN ID of ingress packets being monitored by this VLAN statistics collection instance.
atVlanStatCollectionPortMap {atVlanStatCollectionEntry 3}	A bitwise port map indicating the switch ports being monitored by this VLAN statistics collection instance. The bit position within the string, maps to the port with the same index in dot1dBasePortTable in BRIDGE-MIB. A binary '1' indicates that the port is being monitored by this VLAN statistics collection instance, with a '0' indicating that it is not.

Table 95-26: AT Enterprise MIB - High Level Structure(cont.)

Object		Description
	atVlanStatCollectionIngressPkts {atVlanStatCollectionEntry 4}	The number of ingress packets received and counted by this VLAN statistics collection instance.
	atVlanStatCollectionIngressOctets {atVlanStatCollectionEntry 5}	The number of octets of data received from ingress packets counted by this VLAN statistics collection instance.
	atVlanStatCollectionResetStats {atVlanStatCollectionEntry 6}	When read, this object will always return 2 (false). Setting its value to 1 (true) will cause the matching VLAN statistics collection instance's ingress packets and ingress octet values to be reset to zero.

Other Enterprise MIBs

In general, all objects are supported except where the relevant protocol or feature is either not supported or not applicable to the device. The following MIBs although under the Enterprise Branch (OID 1.3.6.1.4.1) and utilized by AlliedWare Plus products are not within the AlliedTelesis branch of the MIB object tree.

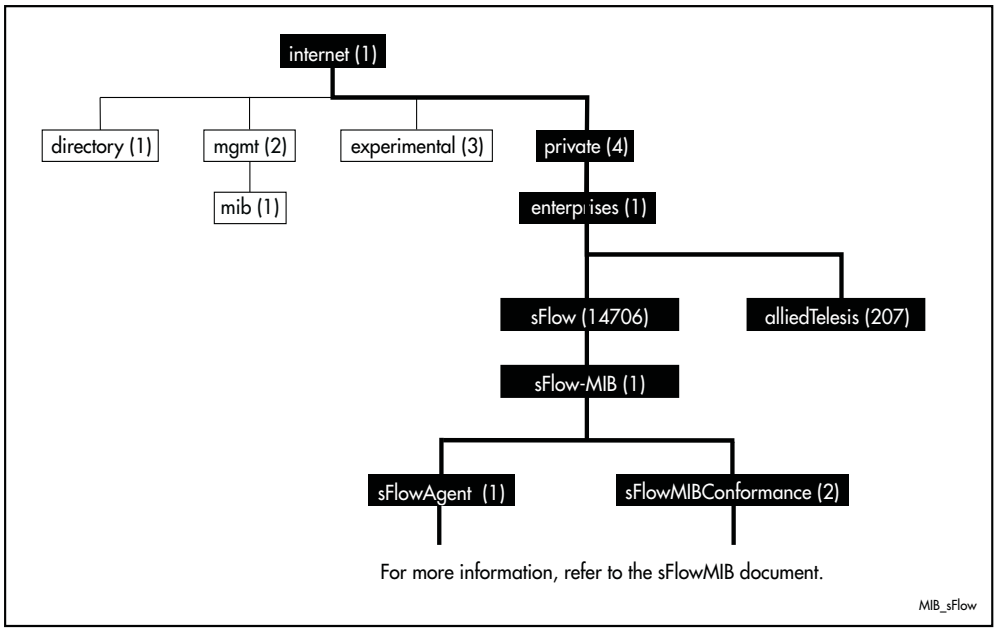
sFlow-MIB

The sFlow-MIB (**Figure 95-9**, and **Table 95-27**) show references to objects for managing the generation and transportation of sFlow data records.

Table 95-27: Objects defined in the sFlow-MIB

MIB Name	Reference / Implementation
sFlow-MIB	All MIB objects are fully supported For more information, see **www.sflow.org/SFLOW-MIB5.txt**

Figure 95-9: The sFlow Statistics MIB tree



Public MIBs

The following table lists the public MIBs supported by the AlliedWare Plus™ Operating System. In general, all objects are supported except where the relevant protocol or feature is either not supported or not applicable to the device. Any variations from the standard are listed.

Public MIBs Supported by AlliedWare Plus™

MIB Name	Reference / Implementation
IANAifType-MIB	**www.iana.org/assignments/ianaiftype-mib**, IANAifType textual convention.
RFC1155-SMI	RFC 1155, *Structure and Identification of Management Information for TCP/IP-based Internets.*
-	RFC 1212, *Concise MIB Definitions*.
RFC1213-MIB	See IP-MIB.
-	RFC 1215, *A Convention for Defining Traps for use with the SNMP*.
-	RFC 1239, *Reassignment of Experimental MIBs to Standard MIBs*.
IP-MIB	The IP MIB tree encompasses IP-MIB, RFC1213-MIB and IP-FORWARD-MIB definitions. The following documents define the components: ■ RFC 1213, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II* ■ RFC 4292, *IP Forwarding Table MIB* ■ RFC 4293, *Management Information Base for the Internet Protocol (IP)* The following objects **are** supported: ■ ipForwarding ■ ipDefaultTTL ■ All ipAddrTable objects except ipAdEntReasmMaxSize ■ All ipNetToPhysicalTable objects except ipNetToPhysicalRowStatus (all read-only) ■ ipCidrRouteNumber ■ All ipCidrRouteTable objects except ipCidrRouteTos All other objects in these MIBs are not supported. Note that an Enterprise version of ipAddressTable objects is provided by atIpAddressTable in AT-IP-MIB. This provides equivalent functionality along with support for primary and secondary IP addresses.
TCP-MIB	RFC 2012, *SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2*.
UDP-MIB	RFC 2013, *SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2*.
IP-FORWARD-MIB	See IP-MIB.
-	RFC 2257, *Agent Extensibility (AgentX) Protocol Version 1*.
SNMP-MPD-MIB	RFC 2572, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*.
SNMP-COMMUNITY-MIB	RFC 2576, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*.
SNMPv2-SMI	RFC 2578, *Structure of Management Information Version 2 (SMIv2)*.
SNMPv2-TC	RFC 2579, *Textual Conventions for SMIv2*.

Public MIBs Supported by AlliedWare Plus™(cont.)

MIB Name	Reference / Implementation
SNMPv2-CONF	RFC 2580, *Conformance Statements for SMIv2*.
P-BRIDGE-MIB	RFC 2674, *Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions*. The following objects are not supported: ■ dot1dTpPortOverflowTable ■ dot1dTrafficClassesEnabled ■ dot1dGmrpStatus ■ dot1dPortCapabilitiesTable ■ dot1dUserPriority ■ dot1dTrafficClassPriority ■ dot1dPortOutboundAccessPriorityTable ■ all objects in the dot1dGarp group ■ all objects in the dot1dGmrp group The following read-write object is implemented as read-only: ■ dot1dPortNumTrafficClasses
Q-BRIDGE-MIB	RFC 2674, *Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions*. The following objects are not supported: ■ dot1qGvrpStatus ■ dot1qFdbId ■ dot1qTpFdbAddress ■ dot1qTpGroupTable ■ dot1qForwardAllTable ■ dot1qForwardUnregisteredTable ■ all objects in the dot1qStatic group ■ dot1qVlanTimeMark ■ dot1qVlanIndex ■ dot1qVlanCurrentEgressPorts ■ dot1qVlanCurrentUntaggedPorts ■ dot1qVlanForbiddenEgressPorts ■ dot1qPortGvrpStatus ■ dot1qPortGvrpFailedRegistrations ■ dot1qPortGvrpLastPduOrigin ■ dot1qPortRestrictedVlanRegistration ■ dot1qPortVlanStatisticsTable ■ dot1qPortVlanHCStatisticsTable ■ dot1qLearningConstraintsTable The following read-write objects are implemented as read-only: ■ dot1qPvid ■ dot1qPortAcceptableFrameTypes
VRRPv3-MIB	RFC 6527, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol Version 3 (VRRPv3)*. All objects with read-write and read-create access are implemented as read-only. RFC 6527 (VRRPv3-MIB) obsoletes RFC 2787 (VRRP-MIB).

Public MIBs Supported by AlliedWare Plus™ (cont.)

MIB Name	Reference / Implementation
HOST-RESOURCES-MIB	RFC 2790, *Host Resources MIB*. The following objects are not supported: ■ hrStorageAllocationFailures ■ All objects in hrDevice ■ All objects in hrSWRun ■ All objects in hrSWRunPerf ■ All objects in hrSWInstalled ■ All objects in hrMIBAdminInfo
SNMPv2-PDU	RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*.
SNMPv2-TM	RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)*.
SNMPv2-MIB	RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*.
POE-MIB	RFC 3621, Power Ethernet MIB. In each of the following objects, if one entry is set then all other entries for the same object in the table are set to the same value. ■ pethMainPseUsageThreshold ■ pethNotificationControlEnable The following objects indicate PSE threshold usage notification: ■ pethMainPowerUsageOnNotification ■ pethMainPowerUsageOffNotification The following read-write object is implemented as read-only: ■ pethPsePortPowerPairs
EtherLike-MIB	RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types*. The following objects are deprecated: ■ dot3StatsEtherChipSet ■ all objects in the dot3Tests group ■ all objects in the dot3Errors group The following read-write object is implemented as read-only: ■ dot3PauseAdminMode

Public MIBs Supported by AlliedWare Plus™ (cont.)

MIB Name	Reference / Implementation
MAU-MIB	RFC 3636, *Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)*. The following objects are not supported: ■ all objects in the dot3RpMauBasicGroup group ■ ifMauTypeListBits ■ ifMauHCFalseCarriers ■ all object identifiers in the dot3MauType group ■ ifMauAutoNegCapabilityBits ■ ifMauAutoNegCapAdvertisedBits ■ ifMauAutoNegCapReceivedBits ■ ifMauAutoNegRemoteFaultAdvertised ■ ifMauAutoNegRemoteFaultReceived ■ all objects in the mauMod group The following objects are deprecated: ■ ifMauTypeList ■ all objects in the dot3BroadMauBasicGroup group ■ ifMauAutoNegCapability ■ ifMauAutoNegCapAdvertised ■ ifMauAutoNegCapReceived The following read-write object is implemented as read-only: ■ ifMauStatus
INET-ADDRESS-MIB	RFC 4001, *Textual Conventions for Internet Network Addresses*.
BRIDGE-MIB	RFC 4188, *Definitions of Managed Objects for Bridges*. The following read-write objects are implemented as read-only: ■ dot1dStpPortEnable ■ dot1dStpPortPathCost The following objects are not supported: ■ dot1dStaticTable ■ dot1dBaseDelayExceededDiscards ■ dot1dBasePortMtuExceededDiscards
RSTP-MIB	RFC 4318, *Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol.* The following read-write objects are implemented as read-only: ■ dot1dStpPortProtocolMigration ■ dot1dStpPortAdminEdgePort ■ dot1dStpPortAdminPointToPoint ■ dot1dStpPortAdminPathCost The following object is deprecated: ■ dot1dStpPathCostDefault
DISMAN-PING-MIB	RFC 4560, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*. The following (lldpLocManAddrTable and lldpConfigManAddrTable) read-write object is implemented as read-only: ■ pingMaxConcurrentRequests You can specify multiple ping operations, but the device only performs one ping at a time (pingMaxConcurrentRequests). The device uses ICMP echo for ping operations (pingImplementationTypeDomains).

Public MIBs Supported by AlliedWare Plus™(cont.)

MIB Name	Reference / Implementation
LLDP-MIB	*IEEE Standard 802.1AB-2005, Section 12, LLDP MIB Definitions*. The following local management address table supports only a single management address per port: ■ lldpConfigManAddrTable
LLDP-EXT-DOT1-MIB	*IEEE Standard 802.1AB-2005, Annex F, IEEE 802.1 Organizationally Specific TLVs., Section F.7.1, IEEE 802.1LLDP extension MIB module*. In each of the following tables, if one entry is set, all other entries in the table are set to the same value. ■ lldpXdot1ConfigVlanNameTxEnable ■ lldpXdot1ConfigProtoVlanTxEnable ■ lldpXdot1ConfigProtocolTxEnable
LLDP-EXT-DOT3-MIB	*IEEE Standard 802.1AB-2005, Annex G, IEEE 802.3 Organizationally Specific TLVs, Section G.7.1, IEEE 802.3 LLDP extension MIB module*
LLDP-EXT-MED-MIB	*ANSI/TIA-1057- 2006, Section 13.3, LLDP-MED MIB Definition*
RIPv2-MIB	RFC1724 - RIP Version 2 MIB Extension

In general, all objects are supported except where the relevant protocol or feature is either not supported or not applicable to the device. The following table lists the private MIBs supported by the AlliedWare Plus™ Operating System. Any variations from the standard are listed.

Chapter 96: LLDP, LLDP-MED and Voice VLAN Introduction and Configuration

Introduction 96.2
Link Layer Discovery Protocol 96.2
LLDP-MED 96.3
Voice VLAN 96.3
LLDP Advertisements 96.4
  Type-Length-Value (TLV) 96.4
  LLDP-MED: Location Identification TLV 96.7
Transmission and Reception 96.8
  LLDP-MED Operation 96.9
Storing LLDP Information 96.10
Configuring LLDP 96.11
  Configure LLDP 96.12
  Configure LLDP-MED 96.14
  Configure Authentication for Voice VLAN 96.19

Introduction

This chapter describes the Link Layer Discovery Protocol (LLDP), LLDP for Media Endpoint Devices (LLDP-MED) and Voice VLAN, and general configuration information for these.

LLDP is designed to be managed with the Simple Network Management Protocol (SNMP), and SNMP-based Network Management Systems (NMS). LLDP can be configured, and the information it provides can be accessed, using either the command line interface or SNMP.

- For detailed descriptions of the commands used to configure LLDP and LLDP-MED, see **Chapter 97, LLDP Commands**.
- For Voice VLAN commands, see **Chapter 19, VLAN Commands**.
- For information about the LLDP and LLDP-MED MIBs, see **"Public MIBs" on page 95.81**.

Link Layer Discovery Protocol

Link Layer Discovery Protocol (LLDP) is a Layer 2 protocol defined by the *IEEE Standard 802.1AB-2005*. This switch supports LLDP as specified in this standard, including *Annex F* and *Annex G*.

LLDP enables Ethernet network devices, such as switches and routers, to transmit and/or receive device-related information to or from directly connected devices on the network, and to store such information learned about other devices. The data sent and received by LLDP is useful for many reasons. The switch can discover neighbors—other devices directly connected to it. Devices can use LLDP to advertise some parts of their Layer 2 configuration to their neighbors, enabling some kinds of misconfiguration to be more easily detected and corrected.

LLDP is a link level ("one hop") protocol; LLDP information can only be sent to and received from devices that are directly connected to each other, or connected via a hub or repeater. Advertised information is not forwarded on to other devices on the network.

The information transmitted in LLDP advertisements flows in one direction only, from one device to its neighbors, and the communication ends there. Transmitted advertisements do not solicit responses, and received advertisements do not solicit acknowledgment.

LLDP operates over physical ports (Layer 2) only. For example, it can be configured on switch ports that belong to static or dynamic aggregated links (channel groups), but not on the aggregated links themselves; and on switch ports that belong to VLANs, but not on the VLANs themselves.

LLDP provides a way for the switch to:

- transmit information about itself to neighbors
- receive device information from neighbors
- store and manage information in an LLDP MIB

Each port can be configured to transmit local information, receive neighbor information, or both.

LLDP defines:

- a set of common advertisements (**"LLDP Advertisements" on page 96.4**)
- a protocol for transmitting and receiving advertisements (**"Transmission and Reception" on page 96.8**)
- a method for storing the information that is contained within received advertisements (**"Storing LLDP Information" on page 96.10**)

Interactions

LLDP has the following interactions with other switch features:

- Spanning tree

  Ports blocked by a spanning tree protocol can still transmit and receive LLDP advertisements.

- 802.1x

  Ports blocked by 802.1x port authorization cannot transmit or receive LLDP advertisements. If LLDP has stored information for a neighbor on the port before it was blocked, this information will eventually time out and be discarded.

- VLAN tagging

  LLDP packets are untagged; they do not contain 802.1Q header information with VLAN identifier and priority tagging.

- Virtual Chassis Stacking (VCStack) resiliency link

  When a port is configured as a VCStack resiliency link port, LLDP does not operate on the port; LLDP neither transmits nor receives advertisements, and any LLDP configuration and data stored for the port, including counters, is discarded.

- Mirror ports

  LLDP does not operate on mirror analyzer ports.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED), is an extension of LLDP used between LAN network connectivity devices, such as this switch, and the media endpoint devices connected to them, such as IP phones. LLDP-MED is specified in *ANSI/TIA-1057-2006*. Of the application types specified in *ANSI/TIA-1057-2006*, the switch supports Application Type 1: Voice.

LLDP-MED uses the LLDP advertisement, transmission and storage mechanisms, but transmits, receives, and stores data specifically related to managing the voice endpoint devices. This includes information about network policy, location, hardware configuration, and, for Power over Ethernet-capable devices, power management.

Voice VLAN

Many IP phones (or other IP voice devices) have two interfaces: one to connect to the network and another that allows a computer or similar device to connect to the network via the IP phone. It is often desirable to treat the voice and data traffic separately so that appropriate Quality of Service (QoS) policies can be applied to each. The Voice VLAN feature uses LLDP-MED to convey configuration information (such as VLAN ID and User Priority tagging, and DiffServ Code Point (DSCP)—**"Differentiated Services Architecture" on page 62.4**) for the voice traffic to the IP phone. In response, the IP phone sends voice traffic according to this configuration. The data traffic coming through the IP phone from the PC is sent with the default configuration, typically untagged with normal priority.

LLDP Advertisements

LLDP transmits advertisements as packets called LLDP Data Units (LLDPDUs). An LLDPDU consists of a set of Type-Length-Value elements (TLV), each of which contains a particular type of information about the device or port transmitting it.

Type-Length-Value (TLV)

A single LLDPDU contains multiple TLVs. TLVs are short information elements that communicate complex data, such as variable length strings, in a standardized format. Each TLV advertises a single type of information, such as its device ID, type, or management addresses. The following table describes fields in a TLV.

Table 96-1: Fields in a Type Length Value element

Field	Description
Type	Identifies the kind of information. It consists of a 7-bit Type code.
Length	Identifies the length of the information. It consists of a 9-bit value that specifies the number of bytes of data in the Value field.
Value	Contains the actual value of the advertised information. This is a variable length data field.

LLDP sends mandatory TLVs in each advertisement; it can also be configured to send one or more optional TLVs, from the following groups:

- Mandatory Base TLVs, included in all LLDP advertisements. See IEEE 802.1AB-2005.
- Optional Base TLVs, which may be included in any LLDP advertisements. See IEEE 802.1AB-2005.
- IEEE 802.1 Organizationally Specific TLVs (802.1 TLVs). See IEEE 802.1AB-2005 Annex F.
- IEEE 802.3 Organizationally Specific TLVs (802.3 TLVs). See IEEE 802.1AB-2005 Annex G.
- LLDP-MED Organizationally Specific TLVs (LLDP-MED TLVs), included in LLDP-MED advertisements. See ANSI/TIA-1057- 2006.

Mandatory and optional TLVs for LLDP and LLDP-MED advertisements are shown in **Table 96-2**.

Table 96-2: TLVs in LLDP advertisements

TLV	Description
Mandatory Base TLVs—IEEE 802.1AB-2005	
Chassis ID	Identifies the device's chassis. On this switch, this is the MAC address of the switch or stack.
Port ID	Identifies the port that transmitted the LLDPDU.
Time To Live (TTL)	Indicates the length of time in seconds for which the information received in the LLDPDU remains valid. If the value is greater than zero, the information is stored in the LLDP remote system MIB. If the value is zero, the information previously received is no longer valid, and is removed from the MIB.
End of LLDPDU	Signals that there are no more TLVs in the LLDPDU.

Table 96-2: TLVs in LLDP advertisements(cont.)

TLV	Description
Optional Base TLVs—IEEE 802.1AB-2005	
Port description	A description of the device's port in alpha-numeric format.
System name	The system's assigned name in alpha-numeric format.
System description	A description of the device in alpha-numeric format. This includes information about the device's hardware and operating system.
System capabilities	The device's router and bridge functions, and whether or not these functions are currently enabled.
Management address	The address of the local LLDP agent. This can be used to obtain information related to the local device.
IEEE 802.1 Organizationally Specific TLVs (802.1 TLVs)—IEEE 802.1AB-2005 Annex F	
Port VLAN	VLAN identifier that the local port associates with untagged or priority tagged frames.
Port & Protocol VLANs	Whether Port & Protocol VLAN is supported and enabled on the port, and the list of Port & Protocol VLAN identifiers.
VLAN Names	List of VLAN names that the port is assigned to.
Protocol IDs	List of protocols that are accessible through the port, for instance: ■ 9000 (Loopback) ■ 00 26 42 42 03 00 00 00 (STP) ■ 00 27 42 42 03 00 00 02 (RSTP) ■ 00 69 42 42 03 00 00 03 (MSTP) ■ 888e01 (802.1x) ■ aa aa 03 00 e0 2b 00 bb (EPSR) ■ 88090101 (LACP) ■ 00540000e302 (Loop protection) ■ 0800 (IPv4) ■ 0806 (ARP) ■ 86dd (IPv6)
IEEE 802.3 Organizationally Specific TLVs (802.3 TLVs)—IEEE 802.1AB-2005 Annex G	
MAC/PHY Configuration/ Status	The current values of the following for the port: ■ Speed and duplex mode auto-negotiation support ■ Auto-negotiation status ■ PMD (physical media dependent) auto-negotiation advertised capability ■ Operational MAU type This TLV is always included in LLDP-MED advertisements.
Power Via MDI	The power-via-MDI capabilities. On devices that are LLDP-MED and PoE-capable, we recommend using the Extended Power-via-MDI TLV instead of this TLV.
Link Aggregation	Whether the link is capable of being aggregated, whether it is currently in an aggregation and if in an aggregation, the port of the aggregation.

Table 96-2: TLVs in LLDP advertisements(cont.)

TLV	Description
Maximum Frame Size	The maximum supported 802.3 frame size that the sending device is capable of receiving—larger frames will be dropped.
LLDP-MED Organizationally Specific TLVs (LLDP-MED TLVs)—ANSI/TIA-1057- 2006	
LLDP-MED Capabilities	Indicates an LLDP-MED capable device, and advertises which LLDP-MED TLVs are supported and enabled, and the device type. For this switch, the device type is Network Connectivity Device. An advertisement containing this TLV is an LLDP-MED advertisement.
Network Policy	Network policy information configured on the port for connected media endpoint devices. The switch supports Application Type 1: Voice, including the following network policy for connected voice devices to use for voice data: ■ Voice VLAN ID ■ Voice VLAN User Priority tagging ■ Voice VLAN Diffserv Code Point (DSCP)
Location Identification	Location information configured for the port, in one or more of the following formats: ■ Civic address ■ Coordinate-based LCI ■ Emergency Location Identification Number (ELIN) For more information, see **"LLDP-MED: Location Identification TLV" on page 96.7**.
Extended Power-via-MDI	For PoE-capable devices, this TLV includes: ■ Power Type field: Power Sourcing Entity (PSE). ■ Power Source field: current power source, either Primary Power Source or Backup Power Source. ■ Power Priority field: power priority configured on the port. ■ Power Value field: In TLVs transmitted by Power Sourcing Equipment (PSE) such as this switch, this advertises the power that the port can supply over a maximum length cable based on its current configuration (that is, it takes into account power losses over the cable). In TLVs received from Powered Device (PD) neighbors, the power value is the power the neighbor requests. Available on devices that are PoE-capable.
Inventory Management TLV Set	Includes the following TLVs, based on the current hardware platform and the software version, identical on every port on the switch: ■ Hardware Revision ■ Firmware Revision ■ Software Revision ■ Serial Number ■ Manufacturer Name ■ Model Name ■ Asset ID On Virtual Chassis Stacking devices, the inventory information is based on the current master.

LLDP-MED: Location Identification TLV

Location information can be configured for each port, and advertised to remote devices, which can then transmit this information in calls; the location associated with voice devices is particularly important for emergency call services. All ports may be configured with the location of the switch, or each port may be configured with the location of the remote voice device connected to it.

The location information for a particular port can be configured using one or more of the following three data formats: coordinate-based, Emergency Location Identification Number (ELIN), and civic address. Up to one location of each type can be assigned to a port.

Location configuration information (LCI) in all configured data formats is transmitted in Location Identification TLVs. When LLDP receives a Location Identification TLV, it updates the remote entry in the LLDP-MED MIB with this information.

Co-ordinate LCI Coordinate-based location data format uses geospatial data, that is, latitude, longitude, and altitude (height or floors), including indications of resolution, with reference to a particular datum: WGS 84, NAD83—North American Vertical Datum of 1988 (NAVD88), or NAD83—Mean Lower Low Water (MLLW). For more information, see *RFC 3825, Dynamic Host Configuration Protocol Option for Coordinate-based Location Configuration Information.*

ELIN LCI Emergency Location Identification Number (ELIN) location data format provides a unique number for each location for Emergency Call Services (ECS). In North America, ELINs are typically 10 digits long; ELINs up to 25 digits are supported.

Civic Address LCI The Civic Address location data format uses common street address format, as described in *RFC4776*.

Transmission and Reception

Table 96-3 describes the LLDP transmission and reception processes. Additional LLDP-MED processes are described in **"LLDP-MED Operation" on page 96.9**.

Table 96-3: LLDP transmission and reception processes

When ...	and ...	then ...
LLDP is enabled	Ports are configured to transmit LLDP advertisements	Regular LLDP advertisements are sent via these ports at intervals determined by the transmit interval. Each advertisement contains local information (from the Local Systems MIB) for all the mandatory TLVs and the optional TLVs that the port is configured to send.
	Ports are configured to receive LLDP advertisements	Information received in advertisements via these ports is stored in the Neighbor table (Remote Systems MIB). This information is retained until it is replaced by a more recent advertisement from the same neighbor or it times out (the TTL elapses).
Local information changes	The transmission delay time has elapsed since the last advertisement was transmitted	New advertisements are sent containing the new set of local information.
Neighbor information changes	Notifications are enabled, and the notification interval has elapsed since the last notification was sent	The SNMP notification (trap) lldpRemTablesChange is sent.
LLDP transmission and reception is disabled on a port.	An LLDP command was used to do this	It transmits a final 'shutdown' LLDPDU with a Time-To-Live (TTL) TLV that has a value of "0". This tells any remote neighboring devices to remove the information associated with this switch from their remote systems MIB. Then it stops transmitting and receiving advertisements. The neighbor information remains in the Remote Systems MIB until it times out.
	A shutdown command was used on the port	It makes a best effort to send a shutdown LLDPDU. Then it stops transmitting and receiving advertisements. The neighbor information remains in the Remote Systems MIB until it times out.
	Something else disabled LLDP, such as Virtual Chassis Stacking (VCStack) failover	It does not send a shutdown LLDPDU. It stops transmitting and receiving advertisements. The neighbor information remains in the Remote Systems MIB until it times out.
	It is enabled again	LLDP reinitializes and resumes transmitting and receiving advertisements after the reinitialization interval has elapsed.
The Neighbor table has 1600 neighbors		It discards any further neighbors.
LLDP receives a LLDPDU or TLV with a detectable error		It discards the incorrect TLV.
LLDP receives a TLV it does not recognize	It contains no basic format errors	It stores it for possible later retrieval by network management (in the unrecognized TLV information table lldpRemUnknownTLVTable in the LLDP MIB).

LLDP-MED Operation

When LLDP is enabled, LLDP-MED is enabled by default, and uses the same LLDP transmission and reception process described in **Table 96-3**. When LLDP receives an advertisement indicating a newly connected LLDP-MED-capable device on a port, it transmits one LLDP-MED advertisement per second via this port, a configurable number of times (the *fast start count*). Thereafter, it sends regular advertisements at the LLDP transmit interval. When the last advertisement for an LLDP-MED-capable device connected to the port times out, it stops sending LLDP-MED advertisements via the port.

If LLDP-MED notifications are enabled for a port, and SNMP traps for LLDP are enabled, LLDP-MED generates a *Topology Change Notification (LLDP-MED lldpXMedTopology ChangeDetected)* when a new LLDP-MED compliant IP telephony device is connected to a port or removed from a port. This notification includes the following information:

- IP Phone Chassis ID and Chassis ID sub-type (IP address)
- LLDP Endpoint Device Class
- Switch Chassis ID (MAC address) and Port ID where the device is attached.

Storing LLDP Information

When an LLDP device receives a valid LLDP advertisement from a neighboring network device, it stores the information in an IEEE-defined Simple Network Management Protocol (SNMP) Management Information Base (MIB).

LLDP stores information in the LLDP MIB defined in Section 12 of the *IEEE Standard 802.1AB-2005*, its extensions defined in *Annex F*, *Annex G*, and ANSI/TIA-1057- 2006, about:

LLDP-EXT-MED-MIB ANSI/TIA-1057- 2006, Section 13.3, LLDP-MED MIB Definition

- Local system information. This is the information that LLDP can transmit in advertisements to its neighbors.
- Remote systems information. This is the data that the device receives in advertisements from its neighbors.
- LLDP configuration. This can be used with SNMP to configure LLDP on the device.
- LLDP statistics. This includes information about LLDP operation on the device, including packet and event counters.

This information can be accessed either via SNMP, or directly using the command line interface.

Local system

Information about your device is called local system information. The LLDP local system MIB maintains this information, which consists of device details, as well as any user-configured information that you have set up for your switch, for example a port description or a management address.

LLDP on this device can store one management address per port, and transmit this in LLDP advertisements. It can store multiple management addresses received from each neighbor.

Remote systems

Information gained from neighboring devices is called remote system information. The LLDP remote systems MIB maintains this information.

The length of time for which neighbor information remains in the LLDP remote systems MIB is determined by the Time-To-Live (TTL) value of received LLDPDUs. When it receives an advertisement from a neighbor, LLDP starts a timer based on the Time To Live (TTL) information in the advertisement. The Time To Live (TTL) information in an advertisement is:
TTL=transmit interval x holdtime multiplier. If the TTL elapses, for instance if the neighbor has been removed, LLDP deletes the neighbor's information from the MIB. This ensures that only valid LLDP information is stored.

Whenever a new neighbor is discovered, or an existing neighbor sends an advertisement with new information that differs from the previous advertisement, for example a new or changed TLV, a remote tables change event is activated. If SNMP notifications are enabled, the notification lldpRemTablesChange is sent.

To prevent the remote systems MIB from using large amounts of memory and possibly affecting the operation of your switch, it limits the number of neighbors it stores information for to 1600. If it is storing information from 1600 neighbors, and detects any more neighbors, it is considered to have too many neighbors, and discards advertisements from the rest. There is no per-port limit to the number of neighbors.

SNMP utilities

An SNMP utility can read the Neighbors table MIB (Remote Systems Data in the LLDP MIB) on a device to find out about the LLDP neighbors it is directly connected to on each port. Then it can read the Neighbors table MIB on each of these neighbors to find out about their neighboring LLDP devices, and so on.

Configuring LLDP

You can configure LLDP on the device using either:

- the command line interface. For detailed descriptions of the commands, see **Chapter 97, LLDP Commands**, or
- SNMP—see **Chapter 95, SNMP MIBs**.

This section includes the following command line interface configuration procedures:

- **"Configure LLDP" on page 96.12**— This procedure includes configuration for LLDP between network connectivity devices; it does not include LLDP-MED. If you are configuring LLDP-MED only, use the following procedure instead of this one.
- **"Configure LLDP-MED" on page 96.14**—This procedure includes the LLDP configuration required to support LLDP-MED, as well as specific LLDP-MED and Voice VLAN configuration.
- **"Configure Authentication for Voice VLAN" on page 96.19**—This procedure includes 802.1X port authentication configuration including dynamic VLAN assignment to be used with LLDP-MED. Use the previous procedure before using this one.

Because LLDP is often used together with SNMP, consider configuring SNMP before you configure LLDP. LLDP transmits large amounts of data about the network. For security reasons, we recommend configuring SNMP for SNMP version 3 only (for read and write access). Remove all SNMPv1 and SNMPv2 configuration. See **Chapter 93, SNMP Introduction**, and **Chapter 94, SNMP Commands**.

Configure LLDP

Use the procedure in **Table 96-4** below to configure LLDP.

Some optional TLVs send information that can be configured by other commands. If LLDP will be configured to send these TLVs, consider whether to configure the corresponding parameters first.

- Port Description. See the **description (interface)** command on page 14.2.
- System Name. See the **hostname** command on page 10.18.

Table 96-4: Configuration procedure for LLDP

Step 1. Enable LLDP		
1.	`awplus#`**configure terminal**	Enter Configuration mode.
2.	`awplus(config)#`**lldp run**	Enable LLDP.
Step 2. Configure ports for LLDP Configure each port to determine whether and which LLDP messages are transmitted and received. If all the ports running LLDP require the same configuration, configure them all together. Otherwise repeat these commands for each port or group of ports that requires a particular configuration.		
3.	`awplus(config)# interface <port-list>`	Enter Interface Configuration mode for the switch ports.
4.	`awplus(config-if)#`**lldp tv-select {[<tlv>]...}** `awplus(config-if)#`**lldp tlv-select all**	By default, the mandatory TLVs are included in LLDP messages. Enable the transmission of one or more optional TLVs through these port as required.
5.	`awplus(config-if)#`**exit**	Return to Global Configuration mode.
6.	`awplus(config)#interface <port-list>`	By default, transmission and reception of LLDP advertisements is enabled on all ports. Enter Interface Configuration mode for any switch ports that should have transmission or reception disabled.
7.	`awplus(config-if)#`**no lldp {[transmit] [receive]}**	Disable transmission and/or reception as required.
8.	`awplus(config-if)#`**exit**	Return to Global Configuration mode.
9.	`awplus(config)#`**exit**	Return to Privileged Exec mode.
Step 3. Check LLDP configuration		
10.	`awplus#`**show lldp** `awplus#`**show lldp interface [<port-list>]** `awplus#`**show lldp local-info [base] [dot1] [dot3] [med] [interface <port-list>]** `awplus#`**show running-config lldp**	Review the LLDP configuration.
Step 4. Monitor LLDP		
11.	`awplus#`**show lldp neighbors** `awplus#`**show lldp neighbors detail** `awplus#`**show lldp statistics** `awplus#`**show lldp statistics interface [<port-list>]**	Monitor LLDP operations and display neighbor information as required.

Table 96-4: Configuration procedure for LLDP (cont.)

Step 5. Advanced LLDP configuration The configuration procedure above and the defaults for other settings suit most networks. Use the following commands for fine tuning if necessary. Timer intervals should be long enough not to create unnecessarily high numbers of advertisements when there are topology changes. However, be aware that if the intervals are long, a neighbor's information can continue to be stored after its information has changed, or after it is disconnected.		
12.	`awplus#`**configure terminal**	Enter Configuration mode.
13.	`awplus(config)#interface <port-list>`	Enter Interface Configuration mode for the switch ports.
14.	`awplus(config-if)#`**lldp management-address <ipaddr>**	Override the default LLDP management address advertised through this port if required. This must be an IPv4 address that is already configured on the device. To see the management address that will be advertised, use the **show lldp local-info** command on page 97.39.
15.	`awplus(config-if)#`**lldp notifications**	By default, SNMP notifications are not transmitted. Enable them for these ports if required. (SNMP LLDP traps (notifications) must also be enabled.)
16.	`awplus(config-if)#`**exit**	Return to Global Configuration mode.
17.	`awplus(config)#`**lldp timer <5-32768>**	The transmit interval determines how often regular LLDP transmits advertisements from each port. The transmit interval must be at least four times the transmission delay. Default: 30 seconds
18.	`awplus(config)#`**lldp notification-interval <5-3600>**	The notification interval determines the minimum interval between sending SNMP notifications (traps). Default: 5 seconds
19.	`awplus(config)#`**lldp tx-delay <1-8192>**	A series of successive changes over a short period of time can trigger the agent to send a large number of LLDPDUs. To prevent this, there is a transmission delay timer. This establishes a minimum length of time that must elapse between successive LLDP transmissions. The transmission delay cannot be greater than a quarter of the transmit interval. Default: **2** seconds
20.	`awplus(config)#`**lldp reinit <1-10>**	Reinitialization delay timer determines the minimum time after disabling LLDP on a port before it can reinitialize. Default: **2** seconds
21.	`awplus(config)#`**lldp holdtime-multiplier <2-10>**	The transmit interval is multiplied by the holdtime multiplier to give the Time To Live (TTL) value that is advertised to neighbors. Default: **4**
22.	`awplus(config)#`**exit**	Return to Privileged Exec mode.
Step 6. Clear data If necessary, you can clear either neighbor information or LLDP statistics for particular ports or all ports.		
23.	`awplus#`**clear lldp table [interface <port-list>]**	Clear the information from the table of neighbor information.
24.	`awplus#`**clear lldp statistics [interface <port-list>]**	Clear LLDP statistics (packet and event counters).

Configure LLDP-MED

Use the procedure in **Table 96-5** to configure LLDP-MED and Voice VLAN for voice devices connected to the switch.

Consider whether you also need to configure:

- Simple Network Management Protocol (**Chapter 94, SNMP Commands**)
- 802.1X port authentication (**Chapter 69, 802.1X Commands**, **Chapter 65, Authentication Commands**, **Chapter 67, AAA Commands**)
- RADIUS server (**Chapter 73, Local RADIUS Server Commands**, or **Chapter 71, RADIUS Commands**)
- Quality of Service (**Chapter 63, QoS Commands**)
- Access Control Lists (**Chapter 58, IPv4 Hardware Access Control List (ACL) Commands** and **Chapter 59, IPv4 Software Access Control List (ACL) Commands**)
- Power over Ethernet (PoE), if the switch supports PoE (**Chapter 25, Power over Ethernet Commands**)

In most cases, configuring LLDP-MED using SNMP or using the CLI command line interface (CLI) described in **Chapter 97, LLDP Commands** has the same effect. However, the effect of configuring location information using SNMP differs from the CLI. When location information is assigned to a port by SNMP and a matching location is not found on the device, then a new location is automatically created and assigned to the specified port. If the location is unset by SNMP later, then the location is removed to prevent accumulating SNMP-set location information. However, if the location is being used for other ports, the automatically created location is not removed until no ports use it. Once it is modified or assigned to other ports by CLI commands, the location remains even after no ports use the location.

Table 96-5: Configuration procedure for Voice VLAN and LLDP-MED

Step 1. Configure a Voice VLAN Create a VLAN for voice data from voice endpoint devices connected to ports on the switch. Specify the network policy for voice data in this voice VLAN. LLDP-MED sends the network policy to voice devices connected to these ports. The voice devices use this network policy to determine the VLAN, priority and DSCP tagging of voice data it transmits.		
1.	`awplus#` **configure terminal**	Enter Global Configuration mode.
2.	`awplus(config)#` **vlan database**	Enter VLAN Database Configuration mode.
3.	`awplus(config-vlan)#` **vlan <vid> [name <vlan-name>] [state {enable|disable}]**	Create a VLAN to be used for the voice data to and from voice devices connected to the switch. By default, the new VLAN is enabled.
4.	`awplus(config-vlan)#` **exit**	Return to global configuration mode.
5.	`awplus(config)# interface` *<port-list>*	Enter interface configuration mode for the ports to be configured with the same network policy. This may be all the switch ports with voice devices connected to them, or a subset if the network policy will differ between ports.

Table 96-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

6.	`awplus(config-if)#` **switchport voice vlan [<vid>|dot1p|dynamic|untagged]**	Specify the VLAN tagging to be used for voice data on these ports. Use the **dynamic** option if the VLAN tagging will be allocated dynamically by a RADIUS server. To configure authentication and dynamic VLAN allocation using the local RADIUS server, see the procedure in **Table 96-6 on page 96.19**. Default: **none**.
7.	`awplus(config-if)#` **switchport voice vlan priority <0-7>**	Specify the priority-tagging that voice endpoint devices should put into their data packets. Default: **5**.
8.	`awplus(config-if)#` **switchport voice dscp <0-63>**	Specify the DSCP value that voice endpoint devices should put into their data packets. Default: **0**.
9.	`awplus(config-if)#` **exit**	Return to global configuration mode.
Step 2. Enable LLDP		
10.	`awplus(config)#` **lldp run**	Enable LLDP on the switch. Default: LLDP is disabled.
11.	`awplus(config)# interface <port-list>`	Enter interface configuration mode for the switch ports LLDP is NOT to run on.
12.	`awplus(config-if)#` **no lldp {[transmit] [receive]}**	Disable transmission or reception on these ports as required. Default: **transmit** and **receive** enabled.
13.	`awplus(config-if)#` **exit**	Return to global configuration mode.
Step 3. Configure LLDP-MED location information Create civic address, coordinate, and/or ELIN locations, and assign them to switch ports.		
14.	`awplus(config)#` **location civic-location identifier <civic-loc-id>**	Specify a civic location ID, and enter configuration mode for this identifier.
15.	`awplus(config-civic)#` **country <country>** `awplus(config-civic)#` **city <city>** `awplus(config-civic)#` **primary-road-name <primary-road-name>** `awplus(config-civic)#` **street-suffix <street-suffix>** `awplus(config-civic)#` **house-number <house-number>** `awplus(config-civic)#` *<other-civic-location-parameters …>*	Specify the civic address location information for the civic address location ID. You must specify a country first, using the upper-case two-letter country code, and then at least one more parameter. For the full set of parameters you can use to specify civic address location, see the **location civic-location configuration** command on page 97.22.
16.	`awplus(config-civic)#` **exit**	Return to global configuration mode.
17.	`awplus(config)#` **location coord-location identifier <coord-loc-id>**	Specify a coordinate location identifier, and enter configuration mode for this identifier.

Table 96-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

18.	`awplus(config-coord)#` **latitude <latitude>** `awplus(config-coord)#` **lat-resolution <lat-resolution>** `awplus(config-coord)#` **longitude <longitude>** `awplus(config-coord)#` **long-resolution <long-resolution>** `awplus(config-coord)#` **altitude <altitude> {meters|floor}** `awplus(config-coord)#` **alt-resolution <alt-resolution>** `awplus(config-coord)#` **datum {wgs84|nad83-navd|nad83-mllw}**	Specify the coordinate location for the coordinate location identifier.
19.	`awplus(config-coord)#` **exit**	Return to global configuration mode.
20.	`awplus(config)#` **location elin-location <elin> identifier <elin-loc-id>**	Specify an ELIN location identifier, and the ELIN for this identifier.
21.	`awplus(config)#` interface *<port-list>*	Enter interface configuration mode for one or more switch ports which require the same location information.
22.	`awplus(config-if)#` **location civic-location-id <civic-loc-id>** `awplus(config-if)#` **location coord-location-id <coord-loc-id>** `awplus(config-if)#` **location elin-location-id <elin-loc-id>**	Assign the civic, coordinate, and/or ELIN location identifier to these ports. LLDP-MED will send the location information associated with a port to the voice endpoint device attached to it.
23.	`awplus(config-if)#` **exit**	Return to global configuration mode.
24.	`awplus(config)#` **exit**	Return to Privileged Exec mode.
Step 4. Review the LLDP configuration		
25.	`awplus#` **show lldp**	Check general LLDP configuration settings.
26.	`awplus#` **show lldp interface [<port-list>]**	Check LLDP configuration for ports.
27.	`awplus#` **show lldp local-info [base] [dot1] [dot3] [med] [interface <port-list>]**	Check the information that may be transmitted in LLDP advertisements from ports.
28.	`awplus#` **show location {civic-location|coord-location|elin-location}** `awplus#` **show location {civic-location|coord-location|elin-location} identifier {<civic-loc-id>|<coord-loc-id>|<elin-loc-id>}** `awplus#` **show location {civic-location|coord-location|elin-location} interface <port-list>**	Check the location information.
29.	`awplus#` **show running-config lldp**	If you want to display all the LLDP configuration, use this command.
Step 5. Monitor LLDP-MED		
30.	`awplus#` **show lldp neighbors [interface <port-list>]** `awplus#` **show lldp neighbors detail [base] [dot1] [dot3] [med] [interface <port-list>]** `awplus#` **show lldp statistics** `awplus#` **show lldp statistics interface [<port-list>]**	Monitor LLDP operation.

Table 96-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

Step 6. Advanced configuration The configuration procedure above and the defaults for other settings suit most networks. Use the following commands for fine tuning if necessary. For information about other advanced configuration for LLDP, including LLDP timers, see **Table 96-4**.		
31.	`awplus#` **configure terminal**	Enter Global Configuration mode.
32.	`awplus(config)#` **lldp faststart-count <1-10>**	By default, when LLDP-MED detects an LLDP-MED capable device on a port, it sends 3 advertisements at 1s intervals. Change the fast start count if required. Default: fast start count is 3
33.	`awplus(config)#` **lldp non-strict-med-tlv-order-check**	By default non-strict order checking for LLDP-MED advertisements is disabled. That is, strict order checking is applied to LLDP-MED advertisements, and LLDP-MED TLVs in non-standard order are discarded. If you require LLDP-MED advertisements with non-standard TLV order to be received and stored, enable non-strict order checking.
34.	`awplus(config)# interface <port-list>`	Enter interface configuration mode for switch ports which will have the same advanced configuration.
35.	`awplus(config-if)#` **lldp management-address <ipaddr>**	Override the default LLDP management address advertised through this port if required. This must be an IPv4 address that is already configured on the device. To see the management address that will be advertised, use the **show lldp local-info** command on page 97.39.
36.	`awplus(config-if)#` **lldp med-notifications**	By default, SNMP notifications are not transmitted. Enable LLDP-MED Topology Change Detected notifications for these ports if required. (SNMP LLDP traps (notifications) must also be enabled.) Default: LLDP-MED notifications disabled
37.	`awplus(config-if)#` **lldp tlv-select {[<tlv>]...}**	Enable the transmission of one or more optional LLDP TLVs in LLDP-MED advertisements through this port as required. The **mac-phy-config** TLV is transmitted in LLDP-MED advertisements whether or not it is enabled by this command. Default: all mandatory TLVs are enabled.
38.	`awplus(config-if)#` **lldp med-tlv-select {[capabilities] [network-policy] [location] [power-management-ext] [inventory-management]}** `awplus(config-if)#` **lldp med-tlv-select all** `awplus(config-if)#` **no lldp med-tlv-select {[capabilities] [network-policy] [location] [power-management-ext] [inventory-management]}** `awplus(config-if)#` **no lldp med-tlv-select all**	Enable or disable the transmission of optional LLDP-MED TLVs in LLDP-MED advertisements through these ports as required. Default: **capabilities**, **network-policy**, **location**, **power-management** are enabled.
39.	`awplus(config-if)#` **exit**	Return to global configuration mode.

Table 96-5: Configuration procedure for Voice VLAN and LLDP-MED(cont.)

40.	`awplus(config)#` **exit**	Return to privileged exec mode.
Step 7. Clear data If necessary, you can clear either neighbor information or LLDP statistics for particular ports or all ports.		
41.	`awplus#` **clear lldp table [interface <port-list>]**	Clear the information from the table of neighbor information.
42.	`awplus#` **clear lldp statistics [interface <port-list>]**	Clear LLDP statistics (packet and event counters).

Configure Authentication for Voice VLAN

Use the following procedure with LLDP-MED and Voice VLAN to configure 802.1X port authentication and dynamic VLAN assignment using the local RADIUS server on the switch to which the voice endpoint devices are connected.

This procedure assumes that you have already:

- configured Voice VLAN and LLDP-MED using the procedure in **Table 96-5 on page 96.14**
- set **switchport voice vlan** to **dynamic** in the above procedure

This procedure configures the local RADIUS server. If your configuration uses one or more remote RADIUS servers instead, set the IP addresses of the remote RADIUS servers using the **radius-server host** command (**Step 2 on page 19**), and skip all the steps that configure the local RADIUS server (**Step 3 on page 19** to **Step 14 on page 20**).

Table 96-6: Configuration procedure for Voice VLAN with RADIUS authentication and dynamic VLAN

Step 1.	**Configure the IP address of the RADIUS host.**	
1.	`awplus#`**configure terminal**	Enter Global Configuration mode.
2.	`awplus(config)#`**radius-server host** 127.0.0.1 key *<key-string>*	Configure the IP address for the RADIUS server to be the local loopback interface address, so that RADIUS requests are sent to the local RADIUS server. Set the key that Network Access Servers (NAS) will need to use to get access to this RADIUS server.
Step 2.	**Enable the local RADIUS server.**	
3.	`awplus(config)#` **radius-server local**	Enter RADIUS Server Configuration mode.
4.	`awplus(config-radsrv)#` **server enable**	Enable the local RADIUS server.
5.	`awplus(config-radsrv)#` **nas** 127.0.0.1 key *<key-string>*	Set the switch as a client device (Network Access Server), to allow it to send authentication requests to the local RADIUS server. Use the same local loopback interface IP address and key as in the **radius-server host** command used in **Step 2 on page 19**.
Step 3.	**Configure a local RADIUS user group for connected PCs.**	
6.	`awplus(config-radsrv)#` **group <user-group-name>**	Create a local RADIUS server user group for PCs connected to the switch, and enter RADIUS Server Group Configuration mode.
7.	`awplus(config-radsrv-group)#` **vlan {<vid>|<vlan-name>}**	Set the VLAN ID for the user group. This will assign the untagged VLAN ID to authenticated ports for PCs connected to the switch. To create multiple user groups for PCs with different VLANs, repeat these two steps.
8.	`awplus(config-radsrv-group)#`**exit**	Return to RADIUS Server Configuration mode.
Step 4.	**Configure a local RADIUS user group for connected phones.**	
9.	`awplus(config-radsrv)#` **group <user-group-name>**	Create a new local RADIUS server user group for phones connected to the switch, and enter RADIUS Server Group Configuration mode.

Table 96-6: Configuration procedure for Voice VLAN with RADIUS authentication and dynamic VLAN(cont.)

10.	`awplus(config-radsrv-group)#` **vlan {<vid>|<vlan-name>}**	Configure the local RADIUS user group for connected phones to use the same VLAN as the PCs in **Step 7**, so that the phones have access to the same untagged VLAN as the PCs.
11.	`awplus(config-radsrv-group)#` **egress-vlan-id** <*vid*> tagged	Set the Egress-VLAN ID attribute for the user group, and set it to send tagged frames. This will assign the tagged VLAN ID to authenticated ports for phones connected to the switch. To create multiple user groups for phones with different VLANs, repeat these two steps.
12.	`awplus(config-radsrv-group)#` **exit**	Return to RADIUS Server Configuration mode.
Step 5. Add users to the local RADIUS server.		
13.	`awplus(config-radsrv)#` user <radius-user-name> password <user-password> group <user-group>	Add RADIUS user names and passwords to the local RADIUS server for authenticating PCs and phones. Assign the corresponding RADIUS server user groups configured in **Step 6** and **Step 9**. See the **user (RADIUS server)** command on page 73.36.
14.	`awplus(config-radsrv)#` **exit**	Return to Global Configuration mode.
Step 6. Create VLANs.		
15.	`awplus(config)#` **vlan database**	Enter VLAN Database Configuration mode.
16.	`awplus(config-vlan)#` **vlan** <*vid-range*>	Create the VLANs corresponding to the VLAN IDs that will be allocated to the authenticated ports, as configured in **Step 7**, **Step 10**, and **Step 11**.
17.	`awplus(config-vlan)#` **exit**	Return to Global Configuration mode.
Step 7. Configure 802.1X port authentication.		
18.	`awplus(config)#` **aaa authentication dot1x** default group radius	Enable 802.1X port authentication and set it to use the default group of RADIUS servers that contains all RADIUS server hosts configured using the **radius-server host** command. In this procedure, the default group consists of the local RADIUS server.
19.	`awplus(config)# interface <port-list>`	Enter interface configuration mode for the ports that have users (PCs and phones) connected to them.
20.	`awplus(config-if)#` **dot1x port-control** auto	Enable 802.1X for port authentication on these ports.
21.	`awplus(config-if)#` **auth host-mode** multi-supplicant	Configure the ports to use multi-supplicant mode for authentication, so that the phone and PC can be dynamically allocated to different VLANs.
22.	`awplus(config-if)#` **auth dynamic-vlan-creation**	Configure the ports to accept dynamic VLAN allocation. In this procedure, the RADIUS server user groups for both the PCs and the phones use the same VLAN (**Step 7** and **Step 10**), so the default rule (**deny**) allows them both the access they need to the port VLAN. For other options, see the **auth dynamic-vlan-creation** command on page 65.6. Default: **deny** differently assigned VLAN IDs.

Table 96-6: Configuration procedure for Voice VLAN with RADIUS authentication and dynamic VLAN(cont.)

23.	`awplus(config-if)#` **exit**	Return to Global Configuration mode.
24.	`awplus(config)#` **exit**	Return to Privileged Exec mode.
Step 8. Review the authentication configuration.		
25.	`awplus#` **show radius local-server group [<user-group-name>]** `awplus#` **show radius local-server nas [<ip-address>]** `awplus#` **show radius local-server user [<user-name>]**	Check the local RADIUS server configuration.
26.	`awplus#` **show vlan {all|brief|dynamic|static|auto|static-ports<1-4094>}**	Check the VLAN configuration.
27.	`awplus#` **show dot1x [all]**	Check the 802.1X authentication configuration.

Chapter 97: LLDP Commands

Introduction 97.2
Command List 97.2
clear lldp statistics 97.2
clear lldp table 97.3
debug lldp 97.4
lldp faststart-count 97.5
lldp holdtime-multiplier 97.6
lldp management-address 97.7
lldp med-notifications 97.8
lldp med-tlv-select 97.9
lldp non-strict-med-tlv-order-check 97.11
lldp notification-interval 97.12
lldp notifications 97.13
lldp port-number-type 97.14
lldp reinit 97.15
lldp run 97.16
lldp timer 97.17
lldp tlv-select 97.18
lldp transmit receive 97.20
lldp tx-delay 97.21
location civic-location configuration 97.22
location civic-location identifier 97.26
location civic-location-id 97.27
location coord-location configuration 97.28
location coord-location identifier 97.30
location coord-location-id 97.31
location elin-location 97.32
location elin-location-id 97.33
show debugging lldp 97.34
show lldp 97.35
show lldp interface 97.37
show lldp local-info 97.39
show lldp neighbors 97.43
show lldp neighbors detail 97.45
show lldp statistics 97.48
show lldp statistics interface 97.49
show location 97.51

Introduction

LLDP and LLDP-MED can be configured using the commands in this chapter, or by using SNMP with the LLDP-MIB and LLDP-EXT-DOT1-MIB (**"Public MIBs" on page 95.81**). The Voice VLAN feature can be configured using commands in **Chapter 19, VLAN Commands**. For more information about LLDP, see **Chapter 96, LLDP, LLDP-MED and Voice VLAN Introduction and Configuration**.

LLDP can transmit a lot of data about the network. Typically, the network information gathered using LLDP is transferred to a Network Management System by SNMP. For security reasons, we recommend using SNMPv3 for this purpose (**Chapter 93, SNMP Introduction**, **Chapter 94, SNMP Commands**).

LLDP operates over physical ports only. For example, it can be configured on switch ports that belong to static or dynamic channel groups, but not on the channel groups themselves.

Command List

This chapter contains an alphabetical list of commands used to configure LLDP.

clear lldp statistics

This command clears all LLDP statistics (packet and event counters) associated with specified ports. If no port list is supplied, LLDP statistics for all ports are cleared.

Syntax `clear lldp statistics [interface <port-list>]`

Parameter	Description
<port-list>	The ports for which the statistics are to be cleared.

Mode Privileged Exec

Examples To clear the LLDP statistics on ports 1.0.1 and 1.0.6, use the command:

```
awplus# clear lldp statistics interface port1.0.1,port1.0.6
```

To clear all LLDP statistics for all ports, use the command:

```
awplus# clear lldp statistics
```

Related Commands **show lldp statistics**
show lldp statistics interface

clear lldp table

This command clears the table of LLDP information received from neighbors through specified ports. If no port list is supplied, neighbor information is cleared for all ports.

Syntax `clear lldp table [interface <port-list>]`

Parameter	Description
<port-list>	The ports for which the neighbor information table is to be cleared.

Mode Privileged Exec

Examples To clear the table of neighbor information received on ports 1.0.1 and 1.0.6, use the command:

```
awplus# clear lldp table interface port1.0.1,port1.0.6
```

To clear the entire table of neighbor information received through all ports, use the command:

```
awplus# clear lldp table
```

Related Commands **show lldp neighbors**

debug lldp

This command enables specific LLDP debug for specified ports. When LLDP debugging is enabled, diagnostic messages are entered into the system log. If no port list is supplied, the specified debugging is enabled for all ports.

The **no** variant of this command disables specific LLDP debug for specified ports. If no port list is supplied, the specified debugging is disabled for all ports.

Syntax

```
debug lldp {[rx][rxpkt][tx][txpkt]} [interface [<port-list>]]
debug lldp operation
no debug lldp {[rx][rxpkt][tx][txpkt]} [interface [<port-list>]]
no debug lldp operation
no debug lldp all
```

Parameter	Description
`rx`	LLDP receive debug.
`rxpkt`	Raw LLDPDUs received in hex format.
`tx`	LLDP transmit debug.
`txpkt`	Raw Tx LLDPDUs transmitted in hex format.
`<port-list>`	The ports for which debug is to be configured.
`operation`	Debug for LLDP internal operation on the switch.
`all`	Disables all LLDP debugging for all ports.

Default By default no debug is enabled for any ports.

Mode Privileged Exec

Examples To enable debugging of LLDP receive on ports 1.0.1 and 1.0.6, use the command:

```
awplus# debug lldp rx interface port1.0.1,port1.0.6
```

To enable debugging of LLDP transmit with packet dump on all ports, use the command:

```
awplus# debug lldp tx txpkt
```

To disable debugging of LLDP receive on ports 1.0.1 and 1.0.6, use the command:

```
awplus# no debug lldp rx interface port1.0.1,port1.0.6
```

To turn off all LLDP debugging on all ports, use the command:

```
awplus# no debug lldp all
```

Related Commands
show debugging lldp
show running-config lldp
terminal monitor

lldp faststart-count

Use this command to set the fast start count for LLDP-MED. The fast start count determines how many fast start advertisements LLDP sends from a port when it starts sending LLDP-MED advertisements from the port, for instance, when it detects a new LLDP-MED capable device.

The **no** variant of this command resets the LLDPD-MED fast start count to the default (3).

Syntax

```
lldp faststart-count <1-10>
no lldp faststart-count
```

Parameter	Description
<1-10>	The number of fast start advertisements to send.

Default The default fast start count is 3.

Mode Global Configuration

Examples To set the fast start count to 5, use the command:

```
awplus# configure terminal
awplus(config)# lldp faststart-count 5
```

To reset the fast start count to the default setting (3), use the command:

```
awplus# configure terminal
awplus(config)# no lldp faststart-count
```

Related Commands show lldp

lldp holdtime-multiplier

This command sets the holdtime multiplier value. The transmit interval is multiplied by the holdtime multiplier to give the Time To Live (TTL) value that is advertised to neighbors.

The **no** variant of this command sets the multiplier back to its default.

Syntax

```
lldp holdtime-multiplier <2-10>
no lldp holdtime-multiplier
```

Parameter	Description
<2-10>	The multiplier factor.

Default The default holdtime multiplier value is 4.

Mode Global Configuration

Usage The Time-To-Live defines the period for which the information advertised to the neighbor is valid. If the Time-To-Live expires before the neighbor receives another update of the information, then the neighbor discards the information from its database.

Examples To set the holdtime multiplier to 2, use the commands:

```
awplus# configure terminal
awplus(config)# lldp holdtime-multiplier 2
```

To set the holdtime multiplier back to its default, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp holdtime-multiplier 2
```

Related Commands show lldp

lldp management-address

This command sets the IPv4 address to be advertised to neighbors (in the Management Address TLV) via the specified ports. This address will override the default address for these ports.

The **no** variant of this command clears the user-configured management IP address advertised to neighbors via the specified ports. The advertised address reverts to the default.

Syntax

```
lldp management-address <ipaddr>
no lldp management-address
```

Parameter	Description
<ipaddr>	The IPv4 address to be advertised to neighbors, in dotted decimal format. This must be one of the IP addresses already configured on the device.

Default The local loopback interface primary IPv4 address if set, else the primary IPv4 interface address of the lowest numbered VLAN the port belongs to, else the MAC address of the device's baseboard if no VLAN IP addresses are configured for the port.

Mode Interface Configuration

Usage To see the management address that will be advertised, use the **show lldp interface** command or **show lldp local-info** command.

Examples To set the management address advertised by ports 1.0.1 and 1.06, to be 192.168.1.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp management-address 192.168.1.6
```

To clear the user-configured management address advertised by ports 1.0.1 and 1.0.6, and revert to using the default address, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp management-address
```

Related Commands

show lldp interface
show lldp local-info

lldp med-notifications

Use this command to enable LLDP to send LLDP-MED Topology Change Detected SNMP notifications relating to the specified ports. The switch sends an SNMP event notification when a new LLDP-MED compliant IP Telephony device is connected to or disconnected from a port on the switch.

Use the **no** variant of this command to disable the sending of LLDP-MED Topology Change Detected notifications relating to the specified ports.

Syntax

```
lldp med-notifications
no lldp med-notifications
```

Default The sending of LLDP-MED notifications is disabled by default.

Mode Interface Configuration

Examples To enable the sending of LLDP-MED Topology Change Detected notifications relating to ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp med-notifications
```

To disable the sending of LLDP-MED notifications relating to ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp med-notifications
```

Related Commands

lldp notification-interval
lldp notifications
snmp-server enable trap
show lldp interface

lldp med-tlv-select

Use this command to enable LLDP-MED Organizationally Specific TLVs for transmission in LLDP advertisements via the specified ports. The LLDP-MED Capabilities TLV must be enabled before any of the other LLDP-MED Organizationally Specific TLVs are enabled.

Use the **no** variant of this command to disable the specified LLDP-MED Organizationally Specific TLVs for transmission in LLDP advertisements via these ports. In order to disable the LLDP-MED Capabilities TLV, you must also disable the rest of these TLVs. Disabling all these TLVs disables LLDP-MED advertisements.

Syntax

```
lldp med-tlv-select {[capabilities] [network-policy] [location]
   [power-management-ext] [inventory-management]}

lldp med-tlv-select all

no lldp med-tlv-select {[capabilities] [network-policy] [location]
   [power-management-ext] [inventory-management]}

no lldp med-tlv-select all
```

Parameter	Description
`capabilities`	LLDP-MED Capabilities TLV. When this is enabled, the MAC/PHY Configuration/Status TLV from IEEE 802.3 Organizationally Specific TLVs is also automatically included in LLDP-MED advertisements, whether or not it has been explicitly enabled by the **lldp tlv-select** command.
`network-policy`	Network Policy TLV. This TLV is transmitted if Voice VLAN parameters have been configured using the commands: ■ **switchport voice dscp** ■ **switchport voice vlan** ■ **switchport voice vlan priority**
`location`	Location Identification TLV. This TLV is transmitted if location information has been configured using the commands: ■ **location elin-location-id** ■ **location civic-location identifier** ■ **location civic-location configuration** ■ **location coord-location identifier** ■ **location coord-location configuration** ■ **location elin-location**
`power-management-ext`	Extended Power-via-MDI TLV. This TLV is transmitted if the port is PoE capable, and PoE is enabled (**power-inline enable** command on page 25.7).
`inventory-management`	Inventory Management TLV Set, including the following TLVs: ■ Hardware Revision ■ Firmware Revision ■ Software Revision ■ Serial Number ■ Manufacturer Name ■ Model Name ■ Asset ID

Parameter(cont.)	Description(cont.)
`all`	All LLDP-MED Organizationally Specific TLVs.

Default By default LLDP-MED Capabilities, Network Policy, Location Identification and Extended Power-via-MDI TLVs are enabled. Therefore, if LLDP is enabled using the **lldp run** command, by default LLDP-MED advertisements are transmitted on ports that detect LLDP-MED neighbors connected to them.

Mode Interface Configuration

Usage LLDP-MED TLVs are only sent in advertisements via a port if there is an LLDP-MED-capable device connected to it. To see whether there are LLDP-MED capable devices connected to the ports, use the **show lldp neighbors** command.

Examples To enable inclusion of the Inventory TLV Set in advertisements transmitted via ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp med-tlv-select inventory-management
```

To exclude the Inventory TLV Set in advertisements transmitted via ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp med-tlv-select inventory-management
```

To disable LLDP-MED advertisements transmitted via ports 1.0.1 and 1.0.6, disable all these TLVs using the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp med-tlv-select all
```

Related Commands
lldp tlv-select
location elin-location-id
location civic-location identifier
location civic-location configuration
location coord-location identifier
location coord-location configuration
location elin-location
show lldp interface
switchport voice dscp
switchport voice vlan
switchport voice vlan priority

lldp non-strict-med-tlv-order-check

Use this command to enable non-strict order checking for LLDP-MED advertisements it receives. That is, use this command to enable LLDP to receive and store TLVs from LLDP-MED advertisements even if they do not use standard TLV order.

Use the **no** variant of this command to disable non-strict order checking for LLDP-MED advertisements, that is, to set strict TLV order checking, so that LLDP discards any LLDP-MED TLVs that occur before the LLDP-MED Capabilities TLV in an advertisement.

Syntax

```
lldp non-strict-med-tlv-order-check
no lldp non-strict-med-tlv-order-check
```

Default By default TLV non-strict order checking for LLDP-MED advertisements is disabled. That is, strict order checking is applied to LLDP-MED advertisements, according to ANSI/TIA-1057, and LLDP-MED TLVs in non-standard order are discarded.

Mode Global Configuration

Usage The ANSI/TIA-1057 specifies standard order for TLVs in LLDP-MED advertisements, and specifies that if LLDP receives LLDP advertisements with non-standard LLDP-MED TLV order, the TLVs in non-standard order should be discarded. This implementation of LLDP-MED follows the standard: it transmits TLVs in the standard order, and by default discards LLDP-MED TLVs that occur before the LLDP-MED Capabilities TLV in an advertisement. However, some implementations of LLDP transmit LLDP-MED advertisements with non-standard TLV order. To receive and store the data from these non-standard advertisements, enable non-strict order checking for LLDP-MED advertisements using this command.

Examples To enable strict TLV order checking, use the commands:

```
awplus# configure terminal
awplus(config)# lldp tlv-order-check
```

To disable strict TLV order checking, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp tlv-order-check
```

Related Commands show running-config lldp

lldp notification-interval

This command sets the notification interval. This is the minimum interval between LLDP SNMP notifications (traps) of each kind (LLDP Remote Tables Change Notification and LLDP-MED Topology Change Notification).

The **no** variant of this command sets the notification interval back to its default.

Syntax

```
lldp notification-interval <5-3600>
no lldp notification-interval
```

Parameter	Description
<5-3600>	The interval in seconds.

Default The default notification interval is 5 seconds.

Mode Global Configuration

Examples To set the notification interval to 20 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# lldp notification-interval 20
```

To set the notification interval back to its default, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp notification-interval
```

Related Commands

lldp notifications
show lldp

lldp notifications

This command enables the sending of LLDP SNMP notifications (traps) relating to specified ports.

The **no** variant of this command disables the sending of LLDP SNMP notifications for specified ports.

Syntax

```
lldp notifications
no lldp notifications
```

Default The sending of LLDP SNMP notifications is disabled by default.

Mode Interface Configuration

Examples To enable sending of LLDP SNMP notifications for ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp notifications
```

To disable sending of LLDP SNMP notifications for ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp notifications
```

Related Commands

lldp notification-interval
show lldp interface
snmp-server enable trap

lldp port-number-type

This command sets the type of port identifier used to enumerate, that is to count, the LLDP MIB local port entries. The LLDP MIB (*IEEE Standard 802.1AB-2005, Section 12, LLDP MIB Definitions*.) requires the port number value to count LLDP local port entries.

This command also enables you to optionally set an interface index to enumerate the LLDP MIB local port entries, if required by your management system.

The **no** variant of this command resets the type of port identifier back to the default setting (number).

Syntax

```
lldp port-number-type [number|ifindex]
no lldp port-number-type
```

Parameter	Description
`number`	Set the type of port identifier to a port number to enumerate the LLDP MIB local port entries.
`ifindex`	Set the type of port identifier to an interface index to enumerate the LLDP MIB local port entries.

Default The default port identifier type is number. The no variant of this command sets the port identifier type to the default.

Mode Global Configuration

Examples To set the type of port identifier used to enumerate LLDP MIB local port entries to port numbers, use the commands:

```
awplus# configure terminal
awplus(config)# lldp port-number-type number
```

To set the type of port identifier used to enumerate LLDP MIB local port entries to interface indexes, use the commands:

```
awplus# configure terminal
awplus(config)# lldp port-number-type ifindex
```

To reset the type of port identifier used to enumerate LLDP MIB local port entries the default (port numbers), use the commands:

```
awplus# configure terminal
awplus(config)# no lldp port-number-type
```

Related Commands **show lldp**

lldp reinit

This command sets the value of the reinitialization delay. This is the minimum time after disabling LLDP on a port before it can reinitialize.

The **no** variant of this command sets the reinitialization delay back to its default setting.

Syntax

```
lldp reinit <1-10>
no lldp reinit
```

Parameter	Description
<1-10>	The delay in seconds.

Default The default reinitialization delay is 2 seconds.

Mode Global Configuration

Examples To set the reinitialization delay to 3 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# lldp reinit 3
```

To set the reinitialization delay back to its default, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp reinit
```

Related Commands show lldp

lldp run

This command enables the operation of LLDP on the device.

The **no** variant of this command disables the operation of LLDP on the device. The LLDP configuration remains unchanged.

Syntax

```
lldp run
no lldp run
```

Default LLDP is disabled by default.

Mode Global Configuration

Examples To enable LLDP operation, use the commands:

```
awplus# configure terminal
awplus(config)# lldp run
```

To disable LLDP operation, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp run
```

Related Commands show lldp

lldp timer

This command sets the value of the transmit interval. This is the interval between regular transmissions of LLDP advertisements.

The **no** variant of this command sets the transmit interval back to its default.

Syntax

```
lldp timer <5-32768>
no lldp timer
```

Parameter	Description
<5-32768>	The transmit interval in seconds. The transmit interval must be at least four times the transmission delay timer (**lldp tx-delay** command).

Default The default transmit interval is 30 seconds.

Mode Global Configuration

Examples To set the transmit interval to 90 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# lldp timer 90
```

To set the transmit interval back to its default, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp timer
```

Related Commands

lldp tx-delay
show lldp

lldp tlv-select

This command enables one or more optional TLVs, or all TLVs, for transmission in LLDP advertisements via the specified ports. The TLVs can be specified in any order; they are placed in LLDP frames in a fixed order (as described in IEEE 802.1AB). The mandatory TLVs (Chassis ID, Port ID, Time To Live, End of LLDPDU) are always included in LLDP advertisements.

In LLDP-MED advertisements the MAC/PHY Configuration/Status TLV will be always be included regardless of whether it is selected by this command.

The **no** variant of this command disables the specified optional TLVs, or all optional TLVs, for transmission in LLDP advertisements via the specified ports.

Syntax

```
lldp tlv-select {[<tlv>]...}
lldp tlv-select all
no lldp tlv-select {[<tlv>]...}
no lldp tlv-select all
```

Parameter	Description
<tlv>	The TLV to transmit in LLDP advertisements. One of these keywords: ■ port-description (specified by the **description (interface)** command on page 14.2) ■ system-name (specified by the **hostname** command on page 10.18) ■ system-description ■ system-capabilities ■ management-address ■ port-vlan ■ port-and-protocol-vlans ■ vlan-names ■ protocol-ids ■ mac-phy-config ■ power-management (Power Via MDI TLV) ■ link-aggregation ■ max-frame-size
`all`	All TLVs.

Default By default no optional TLVs are included in LLDP advertisements. The MAC/PHY Configuration/Status TLV (**mac-phy-config**) is included in LLDP-MED advertisements whether or not it is selected by this command.

Mode Interface Configuration

Examples To include the management-address and system-name TLVs in advertisements transmitted via ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp tlv-select management-address system-
name
```

To include all optional TLVs in advertisements transmitted via ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp tlv-select all
```

To exclude the management-address and system-name TLVs from advertisements transmitted via ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp tlv-select management-address
system-name
```

To exclude all optional TLVs from advertisements transmitted via ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp tlv-select all
```

Related Commands
description (interface)
hostname
lldp med-tlv-select
show lldp interface
show lldp local-info

lldp transmit receive

This command enables transmission and/or reception of LLDP advertisements to or from neighbors through the specified ports.

The **no** variant of this command disables transmission and/or reception of LLDP advertisements through specified ports.

Syntax

```
lldp {[transmit] [receive]}
no lldp {[transmit] [receive]}
```

Parameter	Description
`transmit`	Enable or disable transmission of LLDP advertisements via this port or ports.
`receive`	Enable or disable reception of LLDP advertisements via this port or ports.

Default LLDP advertisement transmission and reception are enabled on all ports by default.

Mode Interface Configuration

Examples To enable transmission of LLDP advertisements on ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp transmit
```

To enable LLDP advertisement transmission and reception on ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# lldp transmit receive
```

To disable LLDP advertisement transmission and reception on ports 1.0.1 and 1.0.6, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.6
awplus(config-if)# no lldp transmit receive
```

Related Commands **show lldp interface**

lldp tx-delay

This command sets the value of the transmission delay timer. This is the minimum time interval between transmitting LLDP advertisements due to a change in LLDP local information.

The **no** variant of this command sets the transmission delay timer back to its default setting.

Syntax

```
lldp tx-delay <1-8192>
no lldp tx-delay
```

Parameter	Description
<1-8192>	The transmission delay in seconds. The transmission delay cannot be greater than a quarter of the transmit interval (**lldp timer** command).

Default The default transmission delay timer is 2 seconds.

Mode Global Configuration

Examples To set the transmission delay timer to 12 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# lldp tx-delay 12
```

To set the transmission delay timer back to its default, use the commands:

```
awplus# configure terminal
awplus(config)# no lldp tx-delay
```

Related Commands

lldp timer
show lldp

location civic-location configuration

Use these commands to configure a civic address location. The country parameter must be specified first, and at least one of the other parameters must be configured before the location can be assigned to a port.

Use the **no** variants of this command to delete civic address parameters from the location.

Syntax

```
country <country>
state <state>
no state
county <county>
no county
city <city>
no city
division <division>
no division
neighborhood <neighborhood>
no neighborhood
street-group <street-group>
no street-group
leading-street-direction <leading-street-direction>
no leading-street-direction
trailing-street-suffix <trailing-street-suffix>
no trailing-street-suffix
street-suffix <street-suffix>
no street-suffix
house-number <house-number>
no house-number
house-number-suffix <house-number-suffix>
no house-number-suffix
landmark <landmark>
no landmark
additional-information <additional-information>
no additional-information
name <name>
no name
postalcode <postalcode>
no postalcode
```

```
building <building>
no building
unit <unit>
no unit
floor <floor>
no floor
room <room>
no room
place-type <place-type>
no place-type
postal-community-name <postal-community-name>
no postal-community-name
post-office-box <post-office-box>
no post-office-box
additional-code <additional-code>
no additional-code
seat <seat>
no seat
primary-road-name <primary-road-name>
no primary-road-name
road-section <road-section>
no road-section
branch-road-name <branch-road-name>
no branch-road-name
sub-branch-road-name <sub-branch-road-name>
no sub-branch-road-name
street-name-pre-modifier <street-name-pre-modifier>
no street-name-pre-modifier
streetname-post-modifier <streetname-post-modifier>
no streetname-post-modifier
```

Parameter	Description
<country>	Upper-case two-letter country code, as specified in *ISO 3166*.
<state>	State (Civic Address (CA) Type 1): national subdivisions (state, canton, region).
<county>	County (CA Type 2): County, parish, gun (JP), district (IN).

Parameter(cont.)	Description(cont.)
<city>	City (CA Type 3): city, township, shi (JP).
<division>	City division (CA Type 4): City division, borough, city district, ward, chou (JP).
<neighborhood>	Neighborhood (CA Type 5): neighborhood, block.
<street-group>	Street group (CA Type 6): group of streets below the neighborhood level.
<leading-street-direction>	Leading street direction (CA Type 16).
<trailing-street-suffix>	Trailing street suffix (CA Type 17).
<street-suffix>	Street suffix (CA Type 18): street suffix or type.
<house-number>	House number (CA Type 19).
<house-number-suffix>	House number suffix (CA Type 20).
<landmark>	Landmark or vanity address (CA Type 21).
<additional-information>	Additional location information (CA Type 22).
<name>	Name (CA Type 23): residence and office occupant.
<postal-code>	Postal/zip code (CA Type 24).
<building>	Building (CA Type 25): structure.
<unit>	Unit (CA Type 26): apartment, suite.
<floor>	Floor (CA Type 27).
<room>	Room (CA Type 28).
<place-type>	Type of place (CA Type 29).
<postal-community-name>	Postal community name (CA Type 30).
<post-office-box>	Post office box (P.O. Box) (CA Type 31).
<additional-code>	Additional code (CA Type 32).
<seat>	Seat (CA Type 33): seat (desk, cubicle, workstation).
<primary-road-name>	Primary road name (CA Type 34).
<road-section>	Road section (CA Type 35).
<branch-road-name>	Branch road name (CA Type 36).
<sub-branch-road-name>	Sub-branch road name (CA Type 37).
<street-name-pre-modifier>	Street name pre-modifier (CA Type 38).
<street-name-post-modifier>	Street name post-modifier (CA Type 39).

Default By default no civic address location information is configured.

Mode Civic Address Location Configuration

Usage The **country** parameter must be configured before any other parameters can be configured; this creates the location. The country parameter cannot be deleted. One or more of the other parameters must be configured before the location can be assigned to a port. The country parameter must be entered as an upper-case two-letter country code, as specified in *ISO 3166*. All other parameters are entered as alpha-numeric strings. Do not configure all the civic address parameters (this would generate TLVs that are too long). Configure a subset of these parameters—enough to consistently and precisely identify the location of the device. If the location is to be used for Emergency Call Service (ECS), the particular ECS application may have guidelines for configuring the civic address location. For more information about civic address format, see **"LLDP-MED: Location Identification TLV" on page 96.7**.

To specify the civic address location, use the **location civic-location identifier** command. To delete the civic address location, use the **no** variant of the **location civic-location identifier** command. To assign the civic address location to particular ports, so that it can be advertised in TLVs from those ports, use the command **location civic-location-id** command.

Examples To configure civic address location 1 with location "27 Nazareth Avenue, Christchurch, New Zealand" in civic-address format, use the commands:

```
awplus# configure terminal
awplus(config)# location civic-location identifier 1
awplus(config-civic)# country NZ
awplus(config-civic)# city Christchurch
awplus(config-civic)# primary-road-name Nazareth
awplus(config-civic)# street-suffix Avenue
awplus(config-civic)# house-number 27
```

Related Commands
location civic-location-id
location civic-location identifier
show lldp local-info
show location

location civic-location identifier

Use this command to enter the Civic Address Location Configuration mode to configure the specified location.

Use the **no** variant of this command to delete a civic address location. This also removes the location from any ports it has been assigned to.

Syntax

```
location civic-location identifier <civic-loc-id>
no location civic-location identifier <civic-loc-id>
```

Parameter	Description
<civic-loc-id>	A unique civic address location ID, in the range 1 to 4095.

Default By default there are no civic address locations.

Mode Global Configuration

Usage To configure the location information for this civic address location identifier, use the **location civic-location configuration** command. To associate this civic location identifier with particular ports, use the **location elin-location-id** command.

Up to 400 locations can be configured on the switch for each type of location information, up to a total of 1200 locations.

Examples To enter Civic Address Location Configuration mode for the civic address location with ID 1, use the commands:

```
awplus# configure terminal
awplus(config)# location civic-location identifier 1
awplus(config-civic)#
```

To delete the civic address location with ID 1, use the commands:

```
awplus# configure terminal
awplus(config)# no location civic-location identifier 1
```

Related Commands

location civic-location-id
location civic-location configuration
show location
show running-config lldp

location civic-location-id

Use this command to assign a civic address location to the ports. The civic address location must already exist. This replaces any previous assignment of civic address location for the ports. Up to one location of each type can be assigned to a port.

Use the **no** variant of this command to remove a location identifier from the ports.

Syntax

```
location civic-location-id <civic-loc-id>
no location civic-location-id [<civic-loc-id>]
```

Parameter	Description
<civic-loc-id>	Civic address location ID, in the range 1 to 4095.

Default By default no civic address location is assigned to ports.

Mode Interface Configuration

Usage The civic address location associated with a port can be transmitted in Location Identification TLVs via the port.

Before using this command, create the location using the following commands:

- **location civic-location identifier** command
- **location civic-location configuration** command

If a civic-address location is deleted using the **no** variant of the **location civic-location identifier** command, it is automatically removed from all ports.

Examples To assign the civic address location 1 to port1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# location civic-location-id 1
```

To remove a civic address location from port1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no location civic-location-id
```

Related Commands

lldp med-tlv-select
location civic-location identifier
location civic-location configuration
show location

location coord-location configuration

Use this command to configure a coordinate-based location. All parameters must be configured before assigning this location identifier to a port.

Syntax

```
latitude <latitude>
lat-resolution <lat-resolution>
longitude <longitude>
long-resolution <long-resolution>
altitude <altitude> {meters|floor}
alt-resolution <alt-resolution>
datum {wgs84|nad83-navd|nad83-mllw}
```

Parameter	Description
<lat-resolution>	Latitude resolution, as a number of valid bits, in the range 0 to 34.
<latitude>	Latitude value in degrees in the range -90.0 to 90.0
<long-resolution>	Longitude resolution, as a number of valid bits, in the range 0 to 34.
<longitude>	Longitude value in degrees, in the range -180.0 to 180.0.
<alt-resolution>	Altitude resolution, as a number of valid bits, in the range 0 to 30. A resolution of 0 can be used to indicate an unknown value.
<altitude>	Altitude value, in meters or floors.
`meters`	The altitude value is in meters.
`floors`	The altitude value is in floors.
`datum`	The geodetic system (or datum) that the specified coordinate values are based on.
`wgs84`	World Geodetic System 1984.
`nad83-navd`	North American Datum 1983 - North American Vertical Datum.
`nad83-mllw`	North American Datum 1983 - Mean Lower Low Water vertical datum.

Default By default no coordinate location information is configured.

Mode Coordinate Configuration

Usage Latitude and longitude values are always stored internally, and advertised in the Location Identification TLV, as 34-bit fixed-point binary numbers, with a 25-bit fractional part, irrespective of the number of digits entered by the user. Likewise altitude is stored as a 30-bit fixed point binary number, with an 8-bit fractional part. Because the user-entered decimal values are stored as fixed point binary numbers, they cannot always be

represented exactly—the stored binary number is converted to a decimal number for display in the output of the **show location** command. For example, a user-entered latitude value of "2.77" degrees is displayed as "2.769999980926513671875000".

The **lat-resolution**, **long-resolution**, and **alt-resolution** parameters allow the user to specify the resolution of each coordinate element as the number of valid bits in the internally-stored binary representation of the value. These resolution values can be used by emergency services to define a search area.

To specify the coordinate identifier, use the **location coord-location identifier** command. To remove coordinate information, delete the coordinate location by using the **no** variant of that command. To associate the coordinate location with particular ports, so that it can be advertised in TLVs from those ports, use the **location elin-location-id** command.

Example To configure the location for the White House in Washington DC, which has the coordinates based on the WGS84 datum of 38.89868 degrees North (with 22 bit resolution), 77.03723 degrees West (with 22 bit resolution), and 15 meters height (with 9 bit resolution), use the commands:

```
awplus# configure terminal
awplus(config)# location coord-location identifier 1
awplus(config-coord)# la-resolution 22
awplus(config-coord)# latitude 38.89868
awplus(config-coord)# lo-resolution 22
awplus(config-coord)# longitude -77.03723
awplus(config-coord)# alt-resolution 9
awplus(config-coord)# altitude 15 meters
awplus(config-coord)# datum wgs84
```

Related Commands
location coord-location-id
location coord-location identifier
show lldp local-info
show location

location coord-location identifier

Use this command to enter Coordinate Location Configuration mode for this coordinate location.

Use the **no** variant of this command to delete a coordinate location. This also removes the location from any ports it has been assigned to.

Syntax

```
location coord-location identifier <coord-loc-id>
no location coord-location identifier <coord-loc-id>
```

Parameter	Description
<coord-loc-id>	A unique coordinate location identifier, in the range 1 to 4095.

Default By default there are no coordinate locations.

Mode Global Configuration

Usage Up to 400 locations can be configured on the switch for each type of location information, up to a total of 1200 locations.

To configure this coordinate location, use the **location coord-location configuration** command. To associate this coordinate location with particular ports, so that it can be advertised in TLVs from those ports, use the **location coord-location-id** command.

Examples To enter Coordinate Location Configuration mode to configure the coordinate location with ID 1, use the commands:

```
awplus# configure terminal
awplus(config)# location coord-location identifier 1
awplus(config-coord)#
```

To delete coordinate location 1, use the commands:

```
awplus# configure terminal
awplus(config)# no location coord-location identifier 1
```

Related Commands

location coord-location-id
location coord-location configuration
show lldp local-info
show location

location coord-location-id

Use this command to assign a coordinate location to the ports. The coordinate location must already exist. This replaces any previous assignment of coordinate location for the ports. Up to one location of each type can be assigned to a port.

Use the **no** variant of this command to remove a location from the ports.

Syntax

```
location coord-location-id <coord-loc-id>
no location coord-location-id [<coord-loc-id>]
```

Parameter	Description
<coord-loc-id>	Coordinate location ID, in the range 1 to 4095.

Default By default no coordinate location is assigned to ports.

Mode Interface Configuration

Usage The coordinate location associated with a port can be transmitted in Location Identification TLVs via the port.

Before using this command, configure the location using the following commands:

- **location coord-location identifier** command
- **location coord-location configuration** command

If a coordinate location is deleted using the **no** variant of the **location coord-location identifier** command, it is automatically removed from all ports.

Examples To assign coordinate location 1 to port1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# location coord-location-id 1
```

To remove a coordinate location from port1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no location coord-location-id
```

Related Commands

lldp med-tlv-select
location coord-location identifier
location coord-location configuration
show location

location elin-location

Use this command to create or modify an ELIN location.

Use the **no** variant of this command to delete an ELIN location, and remove it from any ports it has been assigned to.

Syntax

```
location elin-location <elin> identifier <elin-loc-id>
no location elin-location identifier <elin-loc-id>
```

Parameter	Description
<elin>	Emergency Location Identification Number (ELIN) for Emergency Call Service (ECS), in the range 10 to 25 digits long. In North America, ELINs are typically 10 digits long.
<elin-loc-id>	A unique ELIN location identifier, in the range 1 to 4095.

Default By default there are no ELIN location identifiers.

Mode Global Configuration

Usage Up to 400 locations can be configured on the switch for each type of location information, up to a total of 1200 locations.

To assign this ELIN location to particular ports, so that it can be advertised in TLVs from those ports, use the **location elin-location-id** command.

Examples To create a new ELIN location with ID 1, and configure it with ELIN "1234567890", use the commands:

```
awplus# configure terminal
awplus(config)# location elin-location 1234567890 identifier 1
```

To delete existing ELIN location with ID 1, use the commands:

```
awplus# configure terminal
awplus(config)# no location elin-location identifier 1
```

Related Commands

location elin-location-id
show lldp local-info
show location

location elin-location-id

Use this command to assign an ELIN location to the ports. The ELIN location must already exist. This replaces any previous assignment of ELIN location for the ports. Up to one location of each type can be assigned to a port.

Use the **no** variant of this command to remove a location identifier from the ports.

Syntax

```
location elin-location-id <elin-loc-id>
no location elin-location-id [<elin-loc-id>]
```

Parameter	Description
<elin-loc-id>	ELIN location identifier, in the range 1 to 4095.

Default By default no ELIN location is assigned to ports.

Mode Interface Configuration

Usage An ELIN location associated with a port can be transmitted in Location Identification TLVs via the port.

Before using this command, configure the location using the **location elin-location** command.

If an ELIN location is deleted using the **no** variant of one of the **location elin-location** command, it is automatically removed from all ports.

Examples To assign ELIN location 1 to port 1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# location elin-location-id 1
```

To remove an ELIN location from port 1.0.1, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# no location elin-location-id
```

Related Commands

lldp med-tlv-select
location elin-location
show location

show debugging lldp

This command displays LLDP debug settings for specified ports. If no port list is supplied, LLDP debug settings for all ports are displayed.

Syntax `show debugging lldp [interface <port-list>]`

Parameter	Description
<port-list>	The ports for which the LLDP debug settings are shown.

Mode User Exec and Privileged Exec

Examples To display LLDP debug settings for all ports, use the command:

```
awplus# show debugging lldp
```

To display LLDP debug settings for ports 1.0.1 to 1.0.6, use the command:

```
awplus# show debugging lldp interface port1.0.1-1.0.6
```

Output **Figure 97-1: Example output from the show debugging lldp command**

```
LLDP Debug settings:
Debugging for LLDP internal operation is on
Port       Rx    RxPkt   Tx    TxPkt
-----------------------------------
1.0.1      Yes   Yes     No    No
1.0.2      Yes   No      No    No
1.0.3      No    No      No    No
1.0.4      Yes   Yes     Yes   No
1.0.5      Yes   No      Yes   No
1.0.6      Yes   Yes     Yes   Yes
```

Table 97-1: Parameters in the output of the show debugging lldp command

Parameter	Description
Port	Port name.
Rx	Whether debugging of LLDP receive is enabled on the port.
RxPkt	Whether debugging of LLDP receive packet dump is enabled on the port.
Rx	Whether debugging of LLDP transmit is enabled on the port.
RxPkt	Whether debugging of LLDP transmit packet dump is enabled on the port.

Related Commands debug lldp

show lldp

This command displays LLDP status and global configuration settings.

Syntax `show lldp`

Mode User Exec and Privileged Exec

Example To display LLDP status and global configuration settings, use the command:

```
awplus# show lldp
```

Output

Figure 97-2: Example output from the show lldp command

```
awplus# show lldp

LLDP Global Configuration:              [Default Values]
LLDP Status ............... Enabled     [Disabled]
Notification Interval ..... 5 secs      [5]
Tx Timer Interval ......... 30 secs     [30]
Hold-time Multiplier ...... 4           [4]
(Computed TTL value ....... 120 secs)
Reinitialization Delay .... 2 secs      [2]
Tx Delay .................. 2 secs      [2]
Port Number Type........... Ifindex     [Port-Number]
Fast Start Count .......... 5           [3]

LLDP Global Status:
Total Neighbor Count ...... 47
Neighbors table last updated 0 hrs 0 mins 43 secs ago
```

Table 97-2: Parameters in the output of the show lldp command

Parameter	Description
`LLDP Status`	Whether LLDP is enabled. Default is disabled.
`Notification Interval`	Minimum interval between LLDP notifications.
`Tx Timer Interval`	Transmit interval between regular transmissions of LLDP advertisements.
`Hold-time Multiplier`	The holdtime multiplier. The transmit interval is multiplied by the holdtime multiplier to give the Time To Live (TTL) value that is advertised to neighbors.
`Reinitialization Delay`	The reinitialization delay. This is the minimum time after disabling LLDP transmit on a port before it can reinitialize again.
`Tx Delay`	The transmission delay. This is the minimum time interval between transmitting advertisements due to a change in LLDP local information.
`Port Number Type`	The type of port identifier used to enumerate LLDP MIB local port entries, as set by the lldp port-number-type command.
`Fast Start Count`	The number of times fast start advertisements are sent for LLDP-MED.

Table 97-2: Parameters in the output of the show lldp command(cont.)

Parameter	Description
`Total Neighbor Count`	Number of LLDP neighbors discovered on all ports.
`Neighbors table last updated`	The time since the LLDP neighbor table was last updated.

Related Commands

show lldp interface
show running-config lldp

show lldp interface

This command displays LLDP configuration settings for specified ports. If no port list is specified, LLDP configuration for all ports is displayed.

Syntax `show lldp interface [<port-list>]`

Parameter	Description
<port-list>	The ports for which the LLDP configuration settings are to be shown.

Mode User Exec and Privileged Exec

Examples To display LLDP configuration settings for ports 1.0.1 to 1.0.6, use the command:

```
awplus# show lldp interface port1.0.1-1.0.6
```

To display LLDP configuration settings for all ports, use the command:

```
awplus# show lldp interface
```

Output **Figure 97-3: Example output from the show lldp interface command**

```
awplus# show lldp interface port1.0.1-1.0.8
LLDP Port Status and Configuration:

  * = LLDP is inactive on this port because it is a mirror analyser port
  Notification Abbreviations:
    RC = LLDP Remote Tables Change       TC = LLDP-MED Topology Change
  TLV Abbreviations:
    Base:  Pd = Port Description         Sn = System Name
           Sd = System Description       Sc = System Capabilities
           Ma = Management Address
    802.1: Pv = Port VLAN ID             Pp = Port And Protocol VLAN ID
           Vn = VLAN Name                Pi = Protocol Identity
    802.3: Mp = MAC/PHY Config/Status    Po = Power Via MDI (PoE)
           La = Link Aggregation         Mf = Maximum Frame Size
    MED:   Mc = LLDP-MED Capabilities    Np = Network Policy
           Lo = Location Identification  Pe = Extended PoE    In = Inventory

                                          Optional TLVs Enabled for Tx
 Port     Rx/Tx  Notif  Management Addr  Base       802.1    802.3    MED
-------------------------------------------------------------------------------
 1.0.1    Rx Tx  RC --  192.168.100.123  PdSnSdScMa -------- -------- McNpLoPe--
*1.0.2    -- Tx  RC --  192.168.100.123  PdSnSdScMa -------- -------- McNpLoPe--
 1.0.3    Rx Tx  RC --  192.168.100.123  Pd--SdScMa PvPpVnPi -------- McNpLoPe--
 1.0.4    -- --  RC --  192.168.100.123  PdSnSd--Ma -------- -------- McNpLoPe--
 1.0.5    Rx Tx  RC TC  192.168.100.123  PdSnSdScMa PvPpVnPi -------- McNpLoPe--
 1.0.6    Rx Tx  RC TC  192.168.100.123  Pd----ScMa -------- -------- McNpLoPe--
 1.0.7    Rx Tx  -- TC  192.168.100.123  PdSnSdScMa PvPpVnPi MpPoLaMf McNpLoPeIn
 1.0.8    Rx Tx  -- TC  192.168.1.1      PdSn--ScMa PvPpVnPi -------- McNp------
```

Table 97-3: Parameters in the output of the show lldp interface command

Parameter	Description
`Port`	Port name.
`Rx`	Whether reception of LLDP advertisements is enabled on the port.
`Tx`	Whether transmission of LLDP advertisements is enabled on the port.
`Notif`	Whether sending SNMP notification for LLDP is enabled on the port: ■ RM = Remote Tables Change Notification ■ TP = LLDP-MED Topology Change Notification
`Management Addr`	Management address advertised to neighbors.
`Base TLVs Enabled for Tx`	List of optional Base TLVs enabled for transmission: ■ Pd = Port Description ■ Sn =System Name ■ Sd = System Description ■ Sc =System Capabilities ■ Ma = Management Address
`802.1 TLVs Enabled for Tx`	List of optional 802.1 TLVs enabled for transmission: ■ Pv = Port VLAN ID ■ Pp = Port And Protocol VLAN ID ■ Vn = VLAN Name ■ Pi =Protocol Identity
`802.3 TLVs Enabled for Tx`	List of optional 802.3 TLVs enabled for transmission: ■ Mp = MAC/PHY Configuration/Status ■ Po = Power Via MDI (PoE) ■ La = Link Aggregation ■ Mf = Maximum Frame Size
`MED TLVs Enabled for Tx`	List of optional LLDP-MED TLVs enabled for transmission: ■ Mc = LLDP-MED Capabilities ■ Np = Network Policy ■ Lo = Location Information, ■ Pe = Extended Power-Via-MDI ■ In = Inventory

Related Commands

show lldp
show running-config lldp

show lldp local-info

This command displays local LLDP information that can be transmitted through specified ports. If no port list is entered, local LLDP information for all ports is displayed.

Syntax

```
show lldp local-info [base] [dot1] [dot3] [med] [interface
   <port-list>]
```

Parameter	Description
`base`	Information for base TLVs.
`dot1`	Information for 802.1 TLVs.
`dot3`	Information for 802.3 TLVs.
`med`	Information for LLDP-MED TLVs.
<port-list>	The ports for which the local information is to be shown.

Mode User Exec and Privileged Exec

Usage Whether and which local information is transmitted in advertisements via a port depends on:

- whether the port is set to transmit LLDP advertisements (**lldp transmit receive** command)
- which TLVs it is configured to send (**lldp tlv-select** command, **lldp med-tlv-select** command)

Examples To display local information transmitted via port 1.0.1, use the command:

```
awplus# show lldp local-info interface port1.0.1
```

To display local information transmitted via all ports, use the command:

```
awplus# show lldp local-info
```

Output **Figure 97-4: Example output from the show lldp local-info command**

```
LLDP Local Information:

Local port1.0.1:
  Chassis ID Type ................... MAC address
  Chassis ID ........................ 0015.77c9.7453
  Port ID Type ...................... Interface alias
  Port ID ........................... port1.0.1
  TTL ............................... 120
  Port Description .................. [not configured]
  System Name ....................... awplus
  System Description ................ Allied Telesis router/switch, AW+
                                      v5.4.3A
  System Capabilities - Supported .. Bridge, Router
                      - Enabled .... Bridge, Router
  Management Address ................ 192.168.1.6
  Port VLAN ID (PVID) ............... 1
  Port & Protocol VLAN - Supported . Yes
                       - Enabled ... No
                       - VIDs ..... 0
  VLAN Names ........................ default
  Protocol IDs ...................... 9000, 0026424203000000, 888e01, aaaa03,
                                      88090101, 00540000e302, 0800, 0806, 86dd
  MAC/PHY Auto-negotiation ......... Supported, Enabled
       Advertised Capability ....... 1000BaseTFD, 100BaseTXFD, 100BaseTX,
                                      10BaseTFD, 10BaseT
       Operational MAU Type ........ 1000BaseTFD (30)
  Power Via MDI (PoE) ............... Supported, Enabled
       Port Class .................. PSE
       Pair Control Ability ........ Disabled
       Power Class ................. Unknown
  Link Aggregation .................. Supported, Disabled
  Maximum Frame Size ................ 1522
  LLDP-MED Device Type .............. Network Connectivity
  LLDP-MED Capabilities ............. LLDP-MED Capabilities, Network Policy,
                                      Location Identification,
                                      Extended Power - PSE, Inventory
  Network Policy .................... [not configured]
  Location Identification .......... Civic Address
       Country Code ................ NZ
       City ........................ Christchurch
       Street Suffix ............... Avenue
       House Number ................ 27
       Primary Road Name ........... Nazareth
  Location Identification .......... ELIN
       ELIN ........................ 123456789012
  Extended Power Via MDI (PoE) ..... PSE
       Power Source ................ Primary Power
       Power Priority .............. Low
       Power Value ................. 4.4 Watts
  Inventory Management:
       Hardware Revision ........... A-0
       Firmware Revision ........... 1.1.0
       Software Revision ........... v5.4.3A
       Serial Number ............... G1Q78900B
       Manufacturer Name ........... Allied Telesis Inc.
       Model Name .................. x610-48Ts/XP
       Asset ID .................... [zero length]
```

Table 97-4: Parameters in the output of the show lldp local-info command

Parameter	Description
`Chassis ID Type`	Type of the Chassis ID.
`Chassis ID`	Chassis ID that uniquely identifies the local device.
`Port ID Type`	Type of the Port ID.
`Port ID`	Port ID of the local port through which advertisements are sent.
`TTL`	Number of seconds that the information advertised by the local port remains valid.
`Port Description`	Port description of the local port, as specified by the **description (interface)** command on page 14.2.
`System Name`	System name, as specified by the **hostname** command on page 10.18.
`System Description`	System description.
`System Capabilities (Supported)`	Capabilities that the local port supports.
`System Capabilities (Enabled)`	Enabled capabilities on the local port.
`Management Addresses`	Management address associated with the local port. To change this, use the **lldp management-address** command.
`Port VLAN ID (PVID)`	VLAN identifier associated with untagged or priority tagged frames received via the local port.
`Port & Protocol VLAN (Supported)`	Whether Port & Protocol VLANs (PPV) is supported on the local port.
`Port & Protocol VLAN (Enabled)`	Whether the port is in one or more Port & Protocol VLANs.
`Port & Protocol VLAN (VIDs)`	List of identifiers for Port & Protocol VLANs that the port is in.
`VLAN Names`	List of VLAN names for VLANs that the local port is assigned to.
`Protocol IDs`	List of protocols that are accessible through the local port.
`MAC/PHY Auto-negotiation`	Auto-negotiation support and current status of the 802.3 LAN on the local port.
`Power Via MDI (PoE)`	PoE-capability and current status on the local port.
`Port Class`	Whether the device is a PSE (Power Sourcing Entity) or a PD (Powered Device)

Table 97-4: Parameters in the output of the show lldp local-info command(cont.)

Parameter	Description
Pair Control Ability	Whether power pair selection can be controlled
Power Pairs	Which power pairs are selected for power ("Signal Pairs" or "Spare Pairs") if pair selection can be controlled
Power Class	The power class of the PD device on the port (class 0, 1, 2, 3 or 4)
Link Aggregation	Whether the link is capable of being aggregated and it is currently in an aggregation.
Aggregated Port-ID	Aggregated port identifier.
Maximum Frame Size	The maximum frame size capability of the implemented MAC and PHY.
LLDP-MED Device Type	LLDP-MED device type
LLDP-MED Capabilities	Capabilities LLDP-MED capabilities supported on the local port.
Network Policy	List of network policies configured on the local port.
VLAN ID	VLAN identifier for the port for the specified application type
Tagged Flag	Whether the VLAN ID is to be used as tagged or untagged
Layer-2 Priority:	Layer 2 User Priority (in the range 0 to 7)
DSCP Value	Diffserv codepoint (in the range 0 to 63)
Location Identification	Location configured on the local port.
Extended Power Via MDI (PoE)	PoE-capability and current status of the PoE parameters for Extended Power-Via-MDI TLV on the local port.
Power Source	The power source the switch currently uses; either primary power or backup power.
Power Priority	The power priority configured on the port; either critical, high or low.
Power Value	The total power the switch can source over a maximum length cable to a PD device on the port. The value shows the power value in Watts from the PD side.
Inventory Management	Inventory information for the device.

Related Commands

description (interface)
hostname
lldp transmit receive

show lldp neighbors

This command displays a summary of information received from neighbors via specified ports. If no port list is supplied, neighbor information for all ports is displayed.

Syntax `show lldp neighbors [interface <port-list>]`

Parameter	Description
<port-list>	The ports for which the neighbor information is to be shown.

Mode User Exec and Privileged Exec

Examples To display neighbor information received via all ports, use the command:

```
awplus# show lldp neighbors
```

To display neighbor information received via ports 1.0.1 and 1.0.6 with LLDP-MED configuration, use the command:

```
awplus# show lldp neighbors interface port1.0.1,port1.0.6
```

Output **Figure 97-5: Example output from the show lldp neighbors command**

```
LLDP Neighbor Information:

Total number of neighbors on these ports .... 4

  System Capability Codes:
    O = Other    P = Repeater   B = Bridge                W = WLAN Access Point
    R = Router   T = Telephone  C = DOCSIS Cable Device   S = Station Only
  LLDP-MED Device Type and Power Source Codes:
    1 = Class I    3 = Class III    PSE  = PoE    Both = PoE&Local   Prim = Primary
    2 = Class II   N = Network Con. Locl = Local  Unkn = Unknown     Back = Backup

Local    Neighbor         Neighbor        Neighbor                 System      MED
Port     Chassis ID       Port ID         Sys Name                 Cap.        Ty Pwr
-------------------------------------------------------------------------------
1.0.1    002d.3044.7ba6   port1.0.2       awplus                   OPBWRTCS
1.0.1    0011.3109.e5c6   port1.0.3       AT-9924 switch/route...  --B-R---
1.0.6    0000.10cf.8590   port3           AR-442S                  --B-R---
1.0.6    00ee.4352.df51   192.168.1.2     Jim's desk phone         --B--T--    3   PSE
```

Table 97-5: Parameters in the output of the show lldp neighbors command

Parameter	Description
`Local Port`	Local port on which the neighbor information was received.
`Neighbor Chassis ID`	Chassis ID that uniquely identifies the neighbor.
`Neighbor Port Name`	Port ID of the neighbor.
`Neighbor Sys Name`	System name of the LLDP neighbor.

Table 97-5: Parameters in the output of the show lldp neighbors command(cont.)

Parameter	Description
`Neighbor Capability`	Capabilities that are supported and enabled on the neighbor.
`System Capability`	System Capabilities of the LLDP neighbor.
`MED Device Type`	LLDP-MED Device class (Class I, II, III or Network Connectivity)
`MED Power Source`	LLDP-MED Power Source

Related Commands **show lldp neighbors detail**

show lldp neighbors detail

This command displays in detail the information received from neighbors via specified ports. If no port list is supplied, detailed neighbor information for all ports is displayed.

Syntax

```
show lldp neighbors detail [base] [dot1] [dot3] [med] [interface
   <port-list>]
```

Parameter	Description
`base`	Information for base TLVs.
`dot1`	Information for 802.1 TLVs.
`dot3`	Information for 803.1 TLVs.
`med`	Information for LLDP-MED TLVs.
<port-list>	The ports for which the neighbor information is to be shown.

Mode User Exec and Privileged Exec

Examples To display detailed neighbor information received via all ports, use the command:

```
awplus# show lldp neighbors detail
```

To display detailed neighbor information received via ports 1.0.1, use the command:

```
awplus# show lldp neighbors detail interface port1.0.1
```

Output **Figure 97-6: Example output from the show lldp neighbors detail command**

```
awplus# show lldp neighbors detail interface port1.0.1
LLDP Detailed Neighbor Information:

Local port1.0.1:
  Neighbors table last updated 0 hrs 0 mins 40 secs ago

  Chassis ID Type ................... MAC address
  Chassis ID ........................ 0004.cd28.8754
  Port ID Type ...................... Interface alias
  Port ID ........................... port1.0.6
    TTL ............................. 120 (secs)
  Port Description .................. [zero length]
  System Name ....................... awplus
  System Description ................ Allied Telesis router/switch, AW+ v5.4.4
  System Capabilities - Supported .. Bridge, Router
                      - Enabled .... Bridge, Router
  Management Addresses .............. 0004.cd28.8754
  Port VLAN ID (PVID) ............... 1
  Port & Protocol VLAN - Supported . Yes
                       - Enabled ... Yes
                       - VIDs ..... 5
  VLAN Names ........................ default, vlan5
  Protocol IDs ...................... 9000, 0026424203000000, 888e01, 8100,
                                      88090101, 00540000e302, 0800, 0806, 86dd
  MAC/PHY Auto-negotiation ......... Supported, Enabled
       Advertised Capability ....... 1000BaseTFD, 100BaseTXFD, 100BaseTX,
                                      10BaseTFD, 10BaseT
       Operational MAU Type ........ 1000BaseTFD (30)
  Power Via MDI (PoE) ............... [not advertised]
  Link Aggregation .................. Supported, Disabled
  Maximum Frame Size ................ 1522 (Octets)
  LLDP-MED Device Type .............. Network Connectivity
  LLDP-MED Capabilities ............. LLDP-MED Capabilities, Network Policy,
                                      Location Identification,
                                      Extended Power - PSE, Inventory
  Network Policy .................... [not advertised]
  Location Identification ........... [not advertised]
  Extended Power Via MDI (PoE) ..... PD
           Power Source ............. PSE
           Power Priority ........... High
           Power Value .............. 4.4 Watts
  Inventory Management:
           Hardware Revision ........ X1-0
           Firmware Revision ........ 1.1.0
           Software Revision ........ v5.4.4
           Serial Number ............ M1NB73008
           Manufacturer Name ........ Allied Telesis Inc.
           Model Name ............... x900-12XT/S
           Asset ID ................. [zero length]
```

Table 97-6: Parameters in the output of the show lldp neighbors detail command

Parameter	Description
`Chassis ID Type`	Type of the Chassis ID.
`Chassis ID`	Chassis ID that uniquely identifies the neighbor.
`Port ID Type`	Type of the Port ID.
`Port ID`	Port ID of the neighbor.
`TTL`	Number of seconds that the information advertised by the neighbor remains valid.
`Port Description`	Port description of the neighbor's port.

Table 97-6: Parameters in the output of the show lldp neighbors detail

Parameter	Description
`System Name`	Neighbor's system name.
`System Description`	Neighbor's system description.
`System Capabilities (Supported)`	Capabilities that the neighbor supports.
`System Capabilities (Enabled)`	Capabilities that are enabled on the neighbor.
`Management Addresses`	List of neighbor's management addresses.
`Port VLAN ID (PVID)`	VLAN identifier associated with untagged or priority tagged frames for the neighbor port.
`Port & Protocol VLAN (Supported)`	Whether Port & Protocol VLAN is supported on the LLDP neighbor.
`Port & Protocol VLAN (Enabled)`	Whether Port & Protocol VLAN is enabled on the LLDP neighbor.
`Port & Protocol VLAN (VIDs)`	List of Port & Protocol VLAN identifiers.
`VLAN Names`	List of names of VLANs that the neighbor's port belongs to.
`Protocol IDs`	List of protocols that are accessible through the neighbor's port.
`MAC/PHY Auto-negotiation`	Auto-negotiation configuration and status
`Power Via MDI (PoE)`	PoE configuration and status of 802.3 Power-Via-MDI TLV
`Link Aggregation`	Link aggregation information
`Maximum Frame Size`	The maximum frame size capability
`LLDP-MED Device Type`	LLDP-MED Device type
`LLDP-MED Capabilities`	LLDP-MED capabilities supported
`Network Policy`	List of network policies
`Location Identification`	Location information
`Extended Power Via MDI (PoE)`	PoE-capability and current status
`Inventory Management`	Inventory information

Related Commands **show lldp neighbors**

show lldp statistics

This command displays the global LLDP statistics (packet and event counters).

Syntax `show lldp statistics`

Mode User Exec and Privileged Exec

Example To display global LLDP statistics information, use the command:

```
awplus# show lldp statistics
```

Output

Figure 97-7: Example output from the show lldp statistics command

```
awplus# show lldp statistics

Global LLDP Packet and Event counters:

  Frames:     Out ................... 345
              In .................... 423
              In Errored ............ 0
              In Dropped ............ 0
  TLVs:       Unrecognized .......... 0
              Discarded ............. 0
  Neighbors:  New Entries ........... 20
              Deleted Entries ....... 20
              Dropped Entries ....... 0
              Entry Age-outs ........ 20
```

Table 97-7: Parameters in the output of the show lldp statistics command

Parameter	Description
`Frames Out`	Number of LLDPDU frames transmitted.
`Frames In`	Number of LLDPDU frames received.
`Frames In Errored`	Number of invalid LLDPDU frames received.
`Frames In Dropped`	Number of LLDPDU frames received and discarded for any reason.
`TLVs Unrecognized`	Number of LLDP TLVs received that are not recognized but the TLV type is in the range of reserved TLV types.
`TLVs Discarded`	Number of LLDP TLVs discarded for any reason.
`Neighbors New Entries`	Number of times the information advertised by neighbors has been inserted into the neighbor table.
`Neighbors Deleted Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table.
`Neighbors Dropped Entries`	Number of times the information advertised by neighbors could not be entered into the neighbor table because of insufficient resources.
`Neighbors Entry Age-outs Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table because the information TTL interval has expired.

Related Commands clear lldp statistics
show lldp statistics interface

show lldp statistics interface

This command displays the LLDP statistics (packet and event counters) for specified ports. If no port list is supplied, LLDP statistics for all ports are displayed.

Syntax `show lldp statistics interface [<port-list>]`

Parameter	Description
<port-list>	The ports for which the statistics are to be shown.

Mode User Exec and Privileged Exec

Examples To display LLDP statistics information for all ports, use the command:

```
awplus# show lldp statistics interface
```

To display LLDP statistics information for ports 1.0.1 and 1.0.6, use the command:

```
awplus# show lldp statistics interface port1.0.1,port1.0.6
```

Output

Figure 97-8: Example output from the show lldp statistics interface command

```
awplus# show lldp statistics interface port1.0.1,port1.0.6

LLDP Packet and Event Counters:

port1.0.1
  Frames:     Out ................... 27
              In .................... 22
              In Errored ............ 0
              In Dropped ............ 0
  TLVs:       Unrecognized .......... 0
              Discarded ............. 0
  Neighbors:  New Entries ........... 3
              Deleted Entries ....... 0
              Dropped Entries ....... 0
              Entry Age-outs ........ 0

port1.0.6
  Frames:     Out ................... 15
              In .................... 18
              In Errored ............ 0
              In Dropped ............ 0
  TLVs:       Unrecognized .......... 0
              Discarded ............. 0
  Neighbors:  New Entries ........... 1
              Deleted Entries ....... 0
              Dropped Entries ....... 0
              Entry Age-outs ........ 0
```

Table 97-8: Parameters in the output of the show lldp statistics interface command

Parameter	Description
`Frames Out`	Number of LLDPDU frames transmitted.
`Frames In`	Number of LLDPDU frames received.

Table 97-8: Parameters in the output of the show lldp statistics interface

Parameter	Description
`Frames In Errored`	Number of invalid LLDPDU frames received.
`Frames In Dropped`	Number of LLDPDU frames received and discarded for any reason.
`TLVs Unrecognized`	Number of LLDP TLVs received that are not recognized but the TLV type is in the range of reserved TLV types.
`TLVs Discarded`	Number of LLDP TLVs discarded for any reason.
`Neighbors New Entries`	Number of times the information advertised by neighbors has been inserted into the neighbor table.
`Neighbors Deleted Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table.
`Neighbors Dropped Entries`	Number of times the information advertised by neighbors could not be entered into the neighbor table because of insufficient resources.
`Neighbors Entry Age-outs Entries`	Number of times the information advertised by neighbors has been removed from the neighbor table because the information TTL interval has expired.

Related Commands

clear lldp statistics
show lldp statistics

show location

Use this command to display selected location information configured on the switch.

Syntax

```
show location {civic-location|coord-location|elin-location}

show location {civic-location|coord-location|elin-location}
   identifier {<civic-loc-id>|<coord-loc-id>|<elin-loc-id>}

show location {civic-location|coord-location|elin-location} interface
   <port-list>
```

Parameter	Description
`civic-location`	Display civic location information.
`coord-location`	Display coordinate location information.
`elin-location`	Display ELIN location information.
<civic-loc-id>	Civic address location identifier, in the range 1 to 4095.
<coord-loc-id>	Coordinate location identifier, in the range 1 to 4095.
<elin-loc-id>	ELIN location identifier, in the range 1 to 4095.
<port-list>	Ports to display information about.

Mode User Exec and Privileged Exec

Examples To display a civic address location configured on port1.0.1, use the command:

```
awplus# show location civic-location interface port1.0.1
```

Figure 97-9: Example output from the show location command

```
awplus# show location civic-location interface port1.0.1
Port     ID   Element Type          Element Value
-------------------------------------------------------------
1.0.1    1    Country               NZ
              City                  Christchurch
              Street-suffix         Avenue
              House-number          27
              Primary-road-name     Nazareth
```

To display coordinate location information configured on the identifier 1, use the command:

```
awplus# show location coord-location identifier 1
```

Figure 97-10: Example output from the show location command

```
awplus# show location coord-location identifier 1
  ID  Element Type             Element Value
------------------------------------------------------------------
  1   Latitude Resolution      15 bits
      Latitude                 38.898648113012313842773437 5 degrees
      Longitude Resolution     15 bits
      Longitude                130.232323229312896728515625 0 degrees
      Altitude Resolution      10 bits
      Altitude                 2.50000000 meters
      Map Datum                WGS 84
```

The coordinate location information displayed may differ from the information entered because it is stored in binary format. For more information, see the **location coord-location configuration** command.

To display all ELIN location information configured on the switch, use the command:

```
awplus# show location elin-location
```

Figure 97-11: Example output from the show location command

```
awplus# show location elin-location
  ID  ELIN
-----------------------------------
  1   1234567890
  2   5432154321
```

Related Commands

location elin-location-id
location civic-location identifier
location civic-location configuration
location coord-location identifier
location coord-location configuration
location elin-location

Chapter 98: SMTP Commands

Command List 98.2
debug mail 98.2
delete mail 98.3
mail 98.4
mail from 98.5
mail smtpserver 98.5
show counter mail 98.6
show mail 98.6
undebug mail 98.7

Command List

This chapter provides an alphabetical reference for commands used to configure SMTP.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

debug mail

This command turns on debugging for sending emails.

The **no** variant of this command turns off debugging for sending emails.

Syntax

```
debug mail
no debug mail
```

Mode Privileged Exec

Examples To turn on debugging for sending emails, use the command:

```
awplus# debug mail
```

To turn off debugging for sending emails, use the command:

```
awplus# no debug mail
```

Related Commands

delete mail
mail
mail from
mail smtpserver
show mail
show counter mail
undebug mail

delete mail

This command deletes mail from the queue.

Syntax `delete mail [mail-id <mail-id>|all]`

Parameter	Description	
`mail-id`	Deletes a single mail from the mail queue.	
	<mail-id>	An unique mail ID number. Use the **show mail** command to display this for an item of mail.
`all`	Delete all the mail in the queue.	

Mode Privileged Exec

Examples To delete a unique mail item `20060912142356.1234` from the queue, use the command:

```
awplus# delete mail 20060912142356.1234
```

To delete all mail from the queue, use the command:

```
awplus# delete mail all
```

Related Commands
debug mail
mail
mail from
mail smtpserver
show mail

mail

This command sends an email using the SMTP protocol. If you specify a file the text inside the file is sent in the message body.

If you do not specify the **to**, **file**, or **subject** parameters, the CLI prompts you for the missing information.

Before you can send mail using this command, you must specify the sending email address using the **mail from** command and a mail server using the **mail smtpserver** command.

Syntax `mail [{to <to>|subject <subject>|file <filename>}]`

Parameter	Description
`to`	The email recipient.
	`<to>` Email address.
`subject`	Description of the subject of this email. Use quote marks when the subject text contains spaces.
	`<subject>` String.
`file`	File to insert as text into the message body.
	`<filename>` String.

Mode Privileged Exec

Example To send an email to `rei@nerv.com` with the subject `dummy plug configuration`, and with the message body inserted from the file `plug.conf` use the command:

```
awplus# mail rei@nerv.com subject dummy plug configuration
        filename plug.conf
```

Related Commands
debug mail
delete mail
mail from
mail smtpserver
show mail
show counter mail

mail from

This command sets an email address for the "mail from" SMTP command. You must specify a sending email address with this command before you can send any email.

Syntax `mail from <from>`

Parameter	Description
<from>	The email address that the mail is sent from.

Mode Global Configuration

Example To set the email address you are sending mail from to "kaji@nerv.com, use the command:

```
awplus(config)# mail from kaji@nerv.com
```

Related Commands
delete mail
mail
mail smtpserver
show mail

mail smtpserver

This command sets the IP address of the SMTP server that your device sends email to. You must specify a mail server with this command before you can send any email.

Syntax `mail smtpserver <ip-address>`

Parameter	Description
<ip-address>	Internet Protocol (IP) Address for the mail server specified.

Mode Global Configuration

Example To specify a mail server at 192.168.0.1, use the command:

```
awplus# mail smtpserver 192.168.0.1
```

Related Commands
debug mail
delete mail
mail
mail from
show mail
show counter mail

show counter mail

This command displays the mail counters.

Syntax `show counter mail`

Mode User Exec and Privileged Exec

Output **Figure 98-1: Example output from the show counter mail command**

```
Mail Client (SMTP) counters
Mails Sent            ......... 0
Mails Sent Fails      ......... 1
```

Table 98-1: Parameters in the output of the show counter mail command

Parameter	Description
`Mails Sent`	The number of emails sent successfully since the last device restart.
`Mails Sent Fails`	The number of emails the device failed to send since the last device restart.

Example To show the emails in the queue use the command:

```
awplus# show counter mail
```

Related Commands
debug mail
delete mail
mail
mail from
show mail

show mail

This command displays the emails in the queue.

Syntax `show mail`

Mode Privileged Exec

Example To display the emails in the queue use the command:

```
awplus# show mail
```

Related Commands
delete mail
mail
show counter mail

undebug mail

This command applies the functionality of the no **debug mail** command on page 98.2.

Chapter 99: RMON Introduction and Configuration

Introduction 99.2
Overview 99.2
RMON Configuration Example 99.3

Introduction

The chapter describes the Remote Network MONitoring (RMON) service on the switch, and describes a configuration example showing how to set up an RMON alarm.

This RMON alarm configuration example described creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch port exceeds a threshold, and creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch drops below a lower threshold.

For detailed information about the commands used to configure RMON, see **Chapter 100, RMON Commands**

RMON is disabled by default in AlliedWare Plus™. No RMON alarms or events are configured.

Overview

The Remote Network MONitoring (RMON) MIB (RFC2819) was developed by the IETF to support monitoring and protocol analysis of LANs with a focus on Layer 1 and 2 information in networks. RMON is an industry standard that provides the functionality in network analyzers.

An RMON implementation operates in a client/server model. Monitoring devices (or 'probes') contain RMON agents that collect information and analyze packets. The probes are servers and the Network Management applications that communicate with them are clients. While agent configuration and data collection uses SNMP, RMON operates differently than SNMP systems:

- Probes have responsibility for data collection and processing, reducing SNMP traffic and reducing processing load for clients.
- Information is only transmitted to the management application when required, not polled.

RMON is mainly used for 'flow-based' monitoring, while SNMP is mainly used for 'device-based' management. RMON data collected deals mainly with traffic patterns on the network, and SNMP data collected usually deals with the status of individual devices on the network.

One disadvantage of flow based monitoring is that remote devices have much more of the management burden, and require more resources. AlliedWare Plus minimizes the management and resources burden by implementing a subset of the RMON MIB group to provide a minimal RMON agent implementation supporting statistics, history, alarms, and events.

The RMON groups supported in AlliedWare Plus™ are:

- **Statistics** - collects Ethernet statistics on a switch port, such as utilization and collisions.
- **History** - collects a history of Ethernet statistics on a switch port.
- **Alarms** - monitor a MIB object for a specified interval, trigger an alarm at a specified value (the '**rising threshold**'), and resets the alarm at another value (the '**falling threshold**'). Alarms are used with events to trigger alarms, which generate logs or SNMP traps.
- **Events** - specify the action to take when an event is triggered by an alarm.

  The action of an event can generate a log or an SNMP trap.

RMON Configuration Example

This configuration example sets up an RMON alarm to create SNMP traps and log messages. This RMON alarm creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch port exceeds a threshold, and creates SNMP traps and log messages when the rate of receipt of Broadcast packets on a switch port drops below a lower threshold.

Step 1: Set up an RMON collection on the switch port that is being monitored.

Use the following commands to configure this functionality:

```
awplus# configure terminal
awplus(config)# interface port1.0.4
awplus(config-if)# rmon collection stats 4
```

This will cause the software to build a table in which it stores statistics relating to the switch port.

Step 2: Define an RMON event that will be called by the alarm when the thresholds are passed.

Create this as a 'trap and log' event, so that both an SNMP trap and a log message will be generated. The trap will be sent to the SNMP community named 'public'.

Use the following command to configure this functionality:

```
awplus# configure terminal
awplus(config)# rmon event 10 log trap public
```

Step 3: Create the RMON alarm.

Every 5 seconds, the alarm checks the broadcast packet counter in RMON collection stats 4. If the change in the value of that counter over the 5 second interval exceeds 5000 (1000 broadcasts per second), the alarm will trigger the event defined in step 2 above.

Additionally, when the rate broadcast falls below 500 broadcasts per 5 seconds, then the alarm will trigger the event defined in step 2 above again.

Use the below command to configure this functionality:

```
awplus# configure terminal
awplus(config)# rmon alarm 5 etherStatsBroadcastPkts.4
                interval 5 delta
                rising-threshold 5000 event 10
                falling-threshold 500 event 10
                alarmstartup 3
```

For the variable '`etherStatsBroadcastPkts.4`' in this command, note that '`.4`' refers to the index number of the RMON collection stats 4 as defined on `port1.0.4`.

So, '`etherStatsBroadcastPkts.4`' refers to 'Received broadcasts' in RMON collection stats 4. Further counters for RMON are defined in section 5 of RFC 1757.

Step 4: Enable RMON traps.

To ensure that the SNMP trap is sent, you need to enabled RMON traps, and you need to define a trap host in SNMP. Use the below commands to configure this functionality:

```
awplus# configure terminal
awplus(config)# snmp-server
awplus(config)# snmp-server enable trap rmon
awplus(config)# snmp-server community public
awplus(config)# snmp-server host 192.168.2.254 version 2c
public
```

Note that the resulting log message will be of the form listed below:

```
RMON [1024]: Alarm Index 5 alarm Rising Threshold 5000 alarm
Value 5117 alarm Rising event Index 10 event description
RMON_SNMP
```

Chapter 100: RMON Commands

Command List 100.2
rmon alarm 100.2
rmon collection history 100.4
rmon collection stats 100.5
rmon event 100.6
show rmon alarm 100.7
show rmon event 100.8
show rmon history 100.9
show rmon statistics 100.10

Command List

This chapter provides an alphabetical reference for commands used to configure Remote Monitoring (RMON).

For an introduction to RMON and an RMON configuration example, see **Chapter 99, RMON Introduction and Configuration**

RMON is disabled by default in AlliedWare Plus™. No RMON alarms or events are configured.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

rmon alarm

Use this command to configure an RMON alarm to monitor the value of an SNMP object, and to trigger specified events when the monitored object crosses specified thresholds.

To specify the action taken when the alarm is triggered, use the event index of an event defined by the **rmon event** command.

Use the **no** variant of this command to remove the alarm configuration.

Note Only alarms for switch port interfaces, not for VLAN interfaces, can be configured.

Syntax

```
rmon alarm <alarm-index> <oid> interval <1-2147483647> {delta|
   absolute} rising-threshold <1-2147483647> event <rising-event-
   index> falling-threshold <1-2147483647> event <falling-event-
   index> alarmstartup {1|2|3} [owner <owner>]

no rmon alarm <alarm-index>
```

Parameter	Description
<alarm-index>	*<1-65535>* Alarm entry index value.
<oid>	The variable SNMP MIB Object Identifier (OID) name to be monitored, in the format etherStatsEntry.field.<*stats-index*>. For example, etherStatsEntry.5.22 is the OID for the etherStatsPkts field in the etherStatsEntry table for the interface defined by the <*stats-index*> 22 in the **rmon collection stats** command.
`interval <1-2147483647>`	Polling interval in seconds.
`delta`	The RMON MIB alarmSampleType: the change in the monitored MIB object value between the beginning and end of the polling interval.
`absolute`	The RMON MIB alarmSampleType: the value of the monitored MIB object.

Parameter(cont.)	Description(cont.)
`rising-threshold <1-2147483647>`	Rising threshold value of the alarm entry in seconds.
`<rising-event-index>`	`<1-65535>` The event to be triggered when the monitored object value reaches the rising threshold value. This is an event index of an event specified by the **rmon event** command.
`falling-threshold <1-2147483647>`	Falling threshold value of the alarm entry in seconds.
`<falling-event-index>`	`<1-65535>` The event to be triggered when the monitored object value reaches the falling threshold value. This is an event index of an event specified by the **rmon event** command.
`alarmstartup {1|2|3}`	Whether RMON can trigger a falling alarm (1), a rising alarm (2) or either (3) when you first start monitoring. See the Usage section for more information.
`owner <owner>`	Arbitrary owner name to identify the alarm entry.

Default By default, there are no alarms.

Mode Global Configuration

Usage RMON alarms have a rising and falling threshold. Once the alarm monitoring is operating, you cannot have a falling alarm unless there has been a rising alarm and vice versa.

However, when you start RMON alarm monitoring, an alarm must be generated without the other type of alarm having first been triggered. The **alarmstartup** parameter allows this. It is used to say whether RMON can generate a rising alarm (1), a falling alarm (2) or either alarm (3) as the first alarm.

Note that the SNMP MIB Object Identifier (OID) indicated in the command syntax with <***oid***> must be specified as a dotted decimal value with the form **etherStatsEntry.field.**<***stats-index***>.

Example To configure an alarm to monitor the change per minute in the etherStatsPkt value for interface 22 (defined by stats-index 22 in the **rmon collection stats** command), to trigger event 2 (defined by the **rmon event** command) when it reaches the rising threshold 400, and to trigger event 3 when it reaches the falling threshold 200, and identify this alarm as belonging to Maria, use the commands:

```
awplus# configure terminal
awplus(config)# rmon alarm 229 etherStatsEntry.22.5 interval 60
delta rising-threshold 400 event 2 falling-
threshold 200 event 3 alarmstartup 3 owner
maria
```

Related Commands **rmon collection stats**
rmon event

rmon collection history

Use this command to create a history statistics control group to store a specified number of snapshots (buckets) of the standard RMON statistics for the switch port, and to collect these statistics at specified intervals. If there is sufficient memory available, then the device will allocate memory for storing the set of buckets that comprise this history control.

Use the **no** variant of this command to remove the specified history control configuration.

Note Only a history for switch port interfaces, not for VLAN interfaces, can be collected.

Syntax

```
rmon collection history <history-index> [buckets <1-65535>]
   [interval <1-3600>] [owner <owner>]

no rmon collection history <history-index>
```

Parameter	Description
`<history-index>`	*<1-65535>* A unique RMON history control entry index value.
`buckets <1-65535>`	Number of requested buckets to store snapshots. Default 50 buckets.
`interval <1-3600>`	Polling interval in seconds. Default 1800 second polling interval.
`owner <owner>`	Owner name to identify the entry.

Default The default interval is 1800 seconds and the default buckets is 50 buckets.

Mode Interface Configuration

Example To create a history statistics control group to store 200 snapshots with an interval of 500 seconds, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# rmon collection history 200 buckets 500
                   interval 600 owner herbert
```

To disable the history statistics control group, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.2
awplus(config-if)# no rmon collection history 200
```

rmon collection stats

Use this command to enable the collection of RMON statistics on a switch port, and assign an index number by which to access these collected statistics.

Use the **no** variant of this command to stop collecting RMON statistics on this switch port.

Note Only statistics for switch port interfaces, not for VLAN interfaces, can be collected.

Syntax

```
rmon collection stats <collection-index> [owner <owner>]
no rmon collection stats <collection-index>
```

Parameter	Description
<collection-index>	*<1-65535>* Give this collection of statistics an index number to uniquely identify it. This is the index to use to access the statistics collected for this switch port.
owner *<owner>*	An arbitrary owner name to identify this statistics collection entry.

Default RMON statistics are not enabled by default.

Mode Interface Configuration

Example To enable the collection of RMON statistics with a statistics index of 200, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# rmon collection stats 200 owner myrtle
```

To to stop collecting RMON statistics, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.3
awplus(config-if)# no rmon collection stats 200
```

rmon event

Use this command to create an event definition for a log or a trap or both. The event index for this event can then be referred to by the **rmon alarm** command.

Use the **no** variant of this command to remove the event definition.

Note Only the events for switch port interfaces, not for VLAN interfaces, can be collected.

Syntax

```
rmon event <event-index> [description <description>|owner <owner>|
   trap <trap>]

rmon event <event-index> [log [description <description>|
   owner <owner>|trap <trap>] ]

rmon event <event-index> [log trap [description <description>|
   owner <owner>] ]

no rmon event <event-index>
```

Parameter	Description
<event-index>	*<1-65535>* Unique event entry index value.
`log`	Log event type.
`trap`	Trap event type.
`log trap`	Log and trap event type.
`description <description>`	Event entry description.
`owner <owner>`	Owner name to identify the entry.

Default No event is configured by default.

Mode Global Configuration

Example To create an event definition for a log with an index of 299, use this command:

```
awplus# configure terminal
awplus(config)# rmon event 299 log description cond3 owner
alfred
```

To to remove the event definition, use the command:

```
awplus# configure terminal
awplus(config)# no rmon event 299
```

Related Commands **rmon alarm**

show rmon alarm

Use this command to display the alarms and threshold configured for the RMON probe.

Note Only the alarms for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon alarm`

Mode User Exec and Privileged Exec

Example To display the alarms and threshold, use this command:

```
awplus# show rmon alarm
```

Related Commands rmon alarm

show rmon event

Use this command to display the events configured for the RMON probe.

Note Only the events for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon event`

Mode User Exec and Privileged Exec

Output **Figure 100-1: Example output from the show rmon event command**

```
awplus#sh rmon event
 event Index = 787
       Description TRAP
        Event type log & trap
          Event community name gopher
       Last Time Sent = 0
       Owner  RMON_SNMP

 event Index = 990
       Description TRAP
        Event type trap
          Event community name teabo
       Last Time Sent = 0
       Owner  RMON_SNMP
```

Note The following etherStats counters are not currently available for Layer 3 interfaces:

- etherStatsBroadcastPkts
- etherStatsCRCAlignErrors
- etherStatsUndersizePkts
- etherStatsOversizePkts
- etherStatsFragments
- etherStatsJabbers
- etherStatsCollisions
- etherStatsPkts64Octets
- etherStatsPkts65to127Octets
- etherStatsPkts128to255Octets
- etherStatsPkts256to511Octets
- etherStatsPkts512to1023Octets
- etherStatsPkts1024to1518Octets

Example To display the events configured for the RMON probe, use this command:

```
awplus# show rmon event
```

Related Commands rmon event

show rmon history

Use this command to display the parameters specified on all the currently defined RMON history collections on the device.

Note Only the history for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon history`

Mode User Exec and Privileged Exec

Output **Figure 100-2: Example output from the show rmon history command**

```
awplus#sh rmon history
 history index = 56
         data source ifindex = 4501
         buckets requested = 34
        buckets granted = 34
        Interval = 2000
        Owner Andrew

 history index = 458
         data source ifindex = 5004
         buckets requested = 400
        buckets granted = 400
        Interval = 1500
        Owner trev
=======================================================
```

Note The following etherStats counters are not currently available for Layer 3 interfaces:

- etherStatsBroadcastPkts
- etherStatsCRCAlignErrors
- etherStatsUndersizePkts
- etherStatsOversizePkts
- etherStatsFragments
- etherStatsJabbers
- etherStatsCollisions
- etherStatsPkts64Octets
- etherStatsPkts65to127Octets
- etherStatsPkts128to255Octets
- etherStatsPkts256to511Octets
- etherStatsPkts512to1023Octets
- etherStatsPkts1024to1518Octets

Example To display the parameters specified on all the currently defined RMON history collections, us the commands:

```
awplus# show rmon history
```

Related Commands rmon collection history

show rmon statistics

Use this command to display the current values of the statistics for all the RMON statistics collections currently defined on the device.

Note Only statistics for switch port interfaces, not for VLAN interfaces, can be shown.

Syntax `show rmon statistics`

Mode User Exec and Privileged Exec

Example To display the current values of the statistics for all the RMON statistics collections, us the commands:

```
awplus# show rmon statistics
```

Output **Figure 100-3: Example output from the show rmon statistics command**

```
awplus#show rmon statistics
    rmon collection index 45
    stats->ifindex = 4501
    input packets 1279340, bytes 85858960, dropped 00, multicast packets 1272100
    output packets 7306090, bytes 268724, multicast packets 7305660 broadcast
packets 290
    rmon collection index 679
    stats->ifindex = 5013
    input packets 00, bytes 00, dropped 00, multicast packets 00
    output packets 8554550, bytes 26777324, multicast packets 8546690 broadcast
packets 7720
```

Note The following etherStats counters are not currently available for Layer 3 interfaces:

- etherStatsBroadcastPkts
- etherStatsCRCAlignErrors
- etherStatsUndersizePkts
- etherStatsOversizePkts
- etherStatsFragments
- etherStatsJabbers
- etherStatsCollisions
- etherStatsPkts64Octets
- etherStatsPkts65to127Octets
- etherStatsPkts128to255Octets
- etherStatsPkts256to511Octets
- etherStatsPkts512to1023Octets
- etherStatsPkts1024to1518Octets

Related Commands rmon collection stats

Chapter 101: Triggers Introduction

Introduction 101.2
Trigger Facility 101.2
Configuring a Trigger 101.2
Troubleshooting Triggers 101.5

Introduction

This chapter provides information about the Trigger facility on this switch. For specific configuration examples, see **Chapter 102, Triggers Configuration**. For detailed descriptions of the commands used to configure triggers, see **Chapter 103, Trigger Commands**.

Trigger Facility

The Trigger facility provides a powerful mechanism for automatic and timed management of your device by automating the execution of commands in response to certain events. For example, you can use triggers to deactivate a service during the weekends, or to collect diagnostic information when the CPU usage is high.

A **trigger** is an ordered sequence of scripts that is executed when a certain event occurs. A **script** is a sequence of commands stored as a plain text file on a file subsystem accessible to the device, such as Flash memory. Each trigger may reference multiple scripts and any script may be used by any trigger. When an event activates a trigger, the trigger executes the scripts associated with it in sequence. One script is executed completely before the next script begins. Various types of triggers are supported, each activated in a different way.

Configuring a Trigger

The following describes the general steps to configure a trigger. For specific configuration examples, see **Chapter 102, Triggers Configuration**.

Step 1: Create a configuration script

Create a configuration script with the commands you would like executed when the trigger conditions are met. To create the configuration script using the CLI, use the command:

```
awplus# edit [<filename>]
```

Alternatively, you can create a script on a PC then load it onto your device using the copy (URL) command.

Step 2: Enter the trigger configuration mode

You must be in the Global Configuration mode to reach the Trigger Configuration mode. Use the command:

```
awplus# configure terminal
```

To create a trigger, and enter its configuration mode, use the command:

```
awplus(config)# trigger <1-250>
```

Step 3: Set the trigger type

The trigger type determines how the trigger is activated. To set the trigger to activate:

« when CPU usage reaches a certain level, use the command:

```
awplus(config-trigger)# type cpu <1-100> [up|down|any]
```

« when the link status of a particular interface changes, use the command:

```
awplus(config-trigger)# type interface <interface>
                        [up|down|any]
```

« when the RAM usage reaches a certain level, use the command:

```
awplus(config-trigger)# type memory <1-100> [up|down|any]
```

« periodically after a set number of minutes, use the command:

```
awplus(config-trigger)# type periodic <1-1440>
```

« when a ping poll identifies that a target device's status has changed, use the command:

```
awplus(config-trigger)# type ping-poll <1-100> {up|down}
```

« if your device reboots, use the command:

```
awplus(config-trigger)# type reboot
```

« at a specific time of the day, use the command:

```
awplus(config-trigger)# type time <hh:mm>
```

Note that a combined limit of 10 triggers of the type periodic and type time can be configured. If you attempt to add more than 10 triggers the following error message is displayed:

```
% Cannot configure more than 10 triggers with the type time or
periodic
```

Step 4: Set the time and days that the trigger can activate on

By default triggers can activate at any time of the day, on all days. If you want your trigger to activate only during a specific time of the day, use the command:

```
awplus(config-trigger)# time {[after <hh:mm:ss>]
                        [before <hh:mm:ss>]}
```

If you want your trigger to activate only on a specific date, use the command:

```
awplus(config-trigger)# day <1-31> <month> <2000-2035>
```

If you want the trigger to activate only on specific days of the week, use the command:

```
awplus(config-trigger)# day <weekday>
```

Note that you can set either a specific date, or specific weekdays, but not both.

Step 5: Specify how often the trigger can activate

By default, triggers can activate an unlimited number of times, as long as the trigger conditions are met. To set a limit on the number of times a trigger can activate, use the command:

```
awplus(config-trigger)# repeat {forever|no|once|yes|
<1-4294967294>}
```

You device maintains two counters that track the number of times a trigger has activated. One counts the total number of times the trigger is activated and is only reset if the device restarts, or when the trigger is destroyed. The other counter tracks the permitted number of repetitions. To reset this counter, use the **repeat** command on page 103.6.

Step 6: Add the script to the trigger

You can add up to five scripts to the trigger. When a trigger is activated, it executes the scripts in sequence, with the lowest numbered script activated first. The first script runs to completion before the next script begins. To add a script, use the command:

```
awplus(config-trigger)# script <1-5> {<filename>}
```

Step 7: Specify a description for the trigger

Specify a description for the trigger, so that you can easily identify the trigger in show commands and log output. Use the command:

```
awplus(config-trigger)# description <description>
```

Step 8: Verify the trigger's configuration

To check the configuration of the trigger, use the command:

```
awplus(config-trigger)# show trigger [<1-250>|counter|
full]
```

Troubleshooting Triggers

You can use the trigger diagnostic mode and trigger debugging to test your triggers and troubleshoot any issues.

Diagnostic mode is set per trigger. In this mode the trigger activates if its trigger conditions are met, but does not run any of its scripts. Your device generates a log message to indicate that the trigger was activated. To place a trigger in diagnostic mode, enter the trigger's configuration mode and use the command:

```
awplus(config-trigger)# test
```

To start debugging for triggers, use the command:

```
awplus(config-trigger)# debug trigger
```

This generates detailed messages about how your device is processing the trigger commands and activating the triggers.

Enabling and Disabling

Triggers are enabled by default. This allows the trigger to activate as soon as its trigger conditions are met. If you need to disable a trigger but do not want to delete the trigger, use the command:

```
awplus(config-trigger)# no active
```

To enable the trigger again, use the command:

```
awplus(config-trigger)# active
```

To delete the trigger, use the command:

```
awplus(config-trigger)# no trigger <1-250>
```

Chapter 102: Triggers Configuration

Introduction 102.2
Restrict Internet Access 102.2
Capture Unusual CPU and RAM Activity 102.4
See Daily Statistics 102.6
Turn Off Power to Port LEDs 102.7
Reduce Power Supplied to Ports 102.9
Capture Show Output and Save to a USB Storage Device 102.11
Load a Release File From a USB Storage Device 102.12

Introduction

The chapter describes how to configure triggers to:

- Restrict Internet Access
- **Capture Unusual CPU and RAM Activity**
- **See Daily Statistics**
- **Turn Off Power to Port LEDs**
- **Capture Show Output and Save to a USB Storage Device**

For more information about triggers, see **Chapter 101, Triggers Introduction**. For detailed descriptions of the commands used to configure triggers, see **Chapter 103, Trigger Commands**.

Restrict Internet Access

In the following configuration the ACME company wants to restrict its employees from accessing popular video sharing websites as this is causing bandwidth problems during work hours. The ACME company is happy for workers to access the site after work hours.

Employee PCs at ACME are on vlan2. Two triggers with associated scripts are needed:

- Trigger 1 activates at 8.30am and runs a script called **shutdown.scp**. This script adds commands to restrict access to the specified sites
- Trigger 2 activates at 5.30pm and runs the script called **open.scp**. This script removes the configuration specified by shutdown.scp

1. Create the **shutdown.scp** script

Create a configuration script using Access Control List commands to restrict users on vlan2 from accessing the specific sites.

2. Create the **open.scp** script

Create a script to remove the ACL configuration specified in the **shutdown.scp** file.

3. Configure trigger 1

To create trigger 1, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 1
```

Set the trigger to activate at 8:30am, by using the command:

```
awplus(config-trigger)# type time 08:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **shutdown.scp** to the trigger:

```
awplus(config-trigger)# script 1 shutdown.scp
```

Specify a helpful description, such as **Stops access to video sharing sites**. Use the command:

```
awplus(config-trigger)# description Stops access to video
                        sharing sites
```

Change to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 2

To create trigger 2, use the command:

```
awplus(config)# trigger 2
```

Set the trigger to activate at 5.30pm:

```
awplus(config-trigger)# type time 17:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **open.scp** to the trigger:

```
awplus(config-trigger)# script 1 open.scp
```

Specify a helpful description, such as **Access allowed to video sharing sites**. Use the command:

```
awplus(config-trigger)# description Access allowed to video
                        sharing sites
```

5. Verify the configuration

To check the configuration of the triggers, use the commands:

```
awplus# show trigger 1

awplus# show trigger 2
```

Capture Unusual CPU and RAM Activity

The following configuration allows you to troubleshoot high CPU or RAM usage by the device. It uses two triggers to capture show output, and places this output in a file.

- Trigger 3 activates the script cpu-usage.scp when CPU usage is over 90% and can activate up to 5 times
- Trigger 4 activates the script ram-usage.scp when RAM usage is over 95%, and can activate up to 10 times

1. Create the cpu-usage.scp configuration script

Create a script with the appropriate show command:

```
awplus# show cpu | redirect showcpu.txt
```

The output of the **show cpu** command has been redirected into a file. It is not possible to display trigger script output on the terminal. Redirecting the command output to a file means it is available for later inspection.

If the trigger activates on more than one occasion the contents of **showcpu.txt** will be overwritten with the latest output. To keep a full record for all activations of this trigger an ASH shell script can be added to the trigger to manage the output of the configuration script. For example:

```
#!/bin/ash
date >> showcpu_bkup.txt
cat showcpu.txt >> showcpu_bkup.txt
```

This script concatenates that date and time of activation and the contents of **showcpu.txt** onto the end of the backup file **showcpu_bkup.txt** in flash memory.

Note that the files may grow large accumulating data and consume available flash memory.

2. Create the ram-usage.scp configuration script

Create a script with the appropriate show command:

```
awplus# show memory | redirect showmem.txt
```

The output of the **show memory** command has been redirected into a file. It is not possible to display trigger script output on the terminal. Redirecting the command output to a file means it is available for later inspection.

If the trigger activates on more than one occasion the contents of **showcpu.txt** will be overwritten with the latest output. To keep a full record for all activations of this trigger an ASH shell script can be added to the trigger to manage the output of the configuration script. For example:

```
#!/bin/ash
date >> showmem_bkup.txt
cat showmem.txt >> showmem_bkup.txt
```

This script concatenates that date and time of activation and the contents of **showmem.scp** onto the end of the backup file **showmem_bkup.scp** in flash memory.

Note that the files may grow large accumulating data and consume available flash memory.

3. Configure trigger 3

To create trigger 3, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 3
```

Set the trigger to activate when CPU usage exceeds 80%:

```
awplus(config-trigger)# type cpu 90 up
```

Add the script **cpu-usage.scp** to the trigger:

```
awplus(config-trigger)# script 1 cpu-usage.scp
```

Return to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 4

To create trigger 4, use the command:

```
awplus(config)# trigger 4
```

Set the trigger to activate when RAM usage exceeds 95%:

```
awplus(config-trigger)# type cpu 95 up
```

Add the script **cpu-usage.scp** to the trigger:

```
awplus(config-trigger)# script 1 ram-usage.scp
```

5. Verify the configuration

To check the configuration of the triggers, use the command:

```
awplus# show trigger 3
awplus# show trigger 4
```

See Daily Statistics

The ACME company has recently set up QoS on its traffic to give traffic different priorities to the ISP. ACME wants to assess how much traffic is dropped with the QoS bandwidths set over the next week. To do this, they want to generate an hourly report on QoS traffic on the first day that this is implemented.

- Trigger 5 activates the script **qos-stats.scp** every 60 minutes.

  The trigger is set to only activate during work hours.

1. Create the **qos-stats.scp** script

Create a configuration script with the appropriate show commands. You can either create the configuration script using the CLI with the **edit** command or create a script on a PC then load it onto your device using the **copy (URL)** command on page 7.16.

2. Configure trigger 5

To create trigger 5, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 5
```

Set the trigger to activate periodically every 60 minutes:

```
awplus(config-trigger)# type periodic 60
```

Set the trigger to activate only during the hours of 8:00am and 6:00pm:

```
awplus(config-trigger)# time after 8:00 before 18:00
```

Add the script **qos-stats.scp** to the trigger:

```
awplus(config-trigger)# script 1 qos-stats.scp
```

3. Verify the configuration

To check the configuration of the trigger, use the command:

```
awplus# show trigger 5
```

Turn Off Power to Port LEDs

The following configuration allows you to conserve power by using the eco-friendly LED (Light Emitting Diode) feature to turn off power to the port LEDs during non-work hours.

See the **ecofriendly led** command for a detailed command description and command examples. See the section **"Save Power with the Eco-Friendly Feature" on page 1.32**.

- Trigger 6 activates at 5.30pm and runs a script called **LEDoff.scp**. This script adds commands to turn off power to all the port LEDs
- Trigger 7 activates at 8.30am and runs the script called **LEDon.scp**. This script removes the configuration specified by **LEDoff.scp**

1. Create the **LEDoff.scp** script

Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
configure terminal
ecofriendly led
exit
exit
!
```

2. Create the **LEDon.scp** script

Create a script to remove the configuration specified in the **LEDoff.scp** file. The configuration script for this example is:

```
!
enable
configure terminal
no ecofriendly led
exit
exit
!
```

3. Configure trigger 6

To create trigger 6, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 6
```

Set the trigger to activate at 5:30pm, by using the command:

```
awplus(config-trigger)# type time 17:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **LEDoff.scp** to the trigger:

```
awplus(config-trigger)# script 1 LEDoff.scp
```

Specify a helpful description, such as **Shutdown power to LEDs**. Use the command:

```
awplus(config-trigger)# description Shutdown power to LEDs
```

Change to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 7

To create trigger 7, use the command:

```
awplus(config)# trigger 9
```

Set the trigger to activate at 8.30am:

```
awplus(config-trigger)# type time 08:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **LEDon.scp** to the trigger:

```
awplus(config-trigger)# script 1 LEDon.scp
```

Specify a helpful description, such as **Turn on power to LEDs**. Use the command:

```
awplus(config-trigger)# description Turn on power to LEDs
```

5. Verify the configuration

To check the configuration of the triggers, use the commands:

```
awplus# show trigger 6
awplus# show trigger 7
```

Reduce Power Supplied to Ports

The following configuration allows you to conserve power by using the eco-friendly LPI (Low Power Idle) feature to reduce power supplied to the ports during non-work hours.

See the **ecofriendly lpi** command for a detailed command description and command examples. See the section **"Save Power with the Eco-Friendly Feature" on page 1.32**.

- Trigger 6 activates at 5.30pm and runs a script called **LPIon.scp**. This script adds commands to reduce power to all the ports.
- Trigger 7 activates at 8.30am and runs the script called **LPIoff.scp**. This script removes the configuration specified by **LPIon.scp**.

1. Create the **LPIon.scp** script

Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
configure terminal
ecofriendly lpi
exit
exit
!
```

2. Create the **LPIoff.scp** script

Create a script to remove the configuration specified in the **LPIon.scp** file. The configuration script for this example is:

```
!
enable
configure terminal
no ecofriendly lpi
exit
exit
!
```

3. Configure trigger 6

To create trigger 6, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 6
```

Set the trigger to activate at 5:30pm, by using the command:

```
awplus(config-trigger)# type time 17:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **LPIon.scp** to the trigger:

```
awplus(config-trigger)# script 1 LPIon.scp
```

Specify a helpful description, such as **Turn on LPI**. Use the command:

```
awplus(config-trigger)# description Turn on LPI
```

Change to Global Configuration mode:

```
awplus(config-trigger)# exit
```

4. Configure trigger 7

To create trigger 7, use the command:

```
awplus(config)# trigger 9
```

Set the trigger to activate at 8.30am:

```
awplus(config-trigger)# type time 08:30
```

Set the trigger to activate on Monday, Tuesday, Wednesday, Thursday and Friday:

```
awplus(config-trigger)# day mon tue wed thur fri
```

Add the script **LPIoff.scp** to the trigger:

```
awplus(config-trigger)# script 1 LPIoff.scp
```

Specify a helpful description, such as **Turn off LPI**. Use the command:

```
awplus(config-trigger)# description Turn off LPI
```

5. Verify the configuration

To check the configuration of the triggers, use the commands:

```
awplus# show trigger 6
awplus# show trigger 7
```

Capture Show Output and Save to a USB Storage Device

The following configuration allows you to automatically capture output from the show tech-support command when a USB storage device is inserted into the switch. It uses a script called by the USB storage device trigger to capture the **show tech-support** output and places this output in a file on the USB storage device.

- Trigger 9 activates the script **shtech-sup.scp** when a USB storage device is inserted in the switch

1. Create the **shtech-sup.scp** script

Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
show tech-support outfile usb:support.txt.gz
exit
end
!
```

2. Configure trigger 9

To create trigger 9, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 9
```

Set the trigger to activate on the insertion of a USB storage device:

```
awplus(config-trigger)# type usb in
```

Add the script **shtech-sup.scp** to the trigger:

```
awplus(config-trigger)# script 1 shtech-sup.scp
```

3. Verify the configuration

To check the configuration of the triggers, use the command:

```
awplus# show trigger 9
```

Load a Release File From a USB Storage Device

The following configuration allows you to automatically load a release file from a USB storage device into Flash memory when a USB storage device is inserted into the switch. It uses a script called by the USB trigger to load the release file from the USB storage device.

Note that you can only specify that the release file is on a USB storage device if there is a backup release file already specified in Flash. See the **boot system backup** command for further information.

Caution **Anyone with physical access to the switch and who knows the name of the release file loaded by the trigger could insert a USB storage device and overwrite the boot configuration in Flash memory.**

- Trigger 11 activates the script **copy.scp** when a USB storage device is inserted in the switch

1. Create the **copy.scp** script

Create a configuration script with the commands that are executed when the trigger conditions are met. You can either create the configuration script using the CLI with the edit command or create a script on a PC then load it onto your device using the copy (URL) command. The configuration script for this example is:

```
!
enable
copy usb flash x510-5.4.4-3.5.rel
wait 5
configure terminal
boot system x510-5.4.4-3.5.rel
exit
end
!
```

2. Configure trigger 11

To create trigger 11, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 11
```

Set the trigger to activate on the insertion of a USB storage device:

```
awplus(config-trigger)# type usb in
```

Add the script **copy.scp** to the trigger:

```
awplus(config-trigger)# script 1 copy.scp
```

Specify a helpful description, such as **Load a release file**. Use the command:

```
awplus(config-trigger)# description Load a release file
```

After a USB storage device has been inserted in the switch, use the following two steps to check the trigger and current boot configuration details.

1. Verify the trigger configuration

To check the configuration of the trigger, use the command:

```
awplus# show trigger 11
```

Example output from this command is shown below:

```
awplus#show trigger 11
Trigger Configuration Details
-------------------------------------------------------------
Trigger ..................... 11
Description ................. Load a release file
Type and details ............ USB (in)
Days ........................ smtwtfs
After ....................... 00:00:00
Before ...................... 23:59:59
Active ...................... Yes
Test ........................ No
Trap ........................ Yes
Repeat ...................... Continuous
Modified .................... Wed Sep 15 16:25:33 2010
Number of activations ....... 1
Last activation ............. Wed Sep 15 16:26:49 2010
Number of scripts ........... 1
  1. copy.scp
  2. <not configured>
  3. <not configured>
  4. <not configured>
  5. <not configured>
-------------------------------------------------------------
```

2. Display the current boot configuration

To display the current boot configuration, use the command:

```
awplus# show boot
```

Example output from this command is shown below:

```
awplus#show boot
Boot configuration
-------------------------------------------------------------
Current software   : x510-5.4.4-3.5.rel
Current boot image : flash:/x510-5.4.4-3.5.rel
Backup  boot image : flash:/x510-5.4.4-3.5.rel
Default boot config: flash:/default.cfg
Current boot config: flash:/atplab.cfg (file exists)
Backup  boot config: flash:/default.cfg (file exists)
```

Chapter 103: Trigger Commands

Command List 103.2
active (trigger) 103.2
day 103.3
debug trigger 103.4
description (trigger) 103.5
repeat 103.6
script 103.7
show debugging trigger 103.9
show running-config trigger 103.9
show trigger 103.10
test 103.15
time (trigger) 103.16
trap 103.18
trigger 103.19
trigger activate 103.20
type atmf node 103.21
type cpu 103.23
type interface 103.24
type memory 103.25
type periodic 103.26
type ping-poll 103.27
type reboot 103.27
type stack disabled-master 103.28
type stack link 103.29
type stack master-fail 103.29
type stack member 103.30
type time 103.31
undebug trigger 103.31

Command List

This chapter provides an alphabetical reference for commands used to configure Triggers. For more information, see **Chapter 101, Triggers Introduction** and **Chapter 102, Triggers Configuration**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

active (trigger)

This command enables a trigger. This allows the trigger to activate when its trigger conditions are met.

The **no** variant of this command disables a trigger. While in this state the trigger cannot activate when its trigger conditions are met.

Syntax

```
active
no active
```

Mode Trigger Configuration

Usage Configure a trigger first before you use this command to activate it. See the **Configuring a Trigger** section in **Chapter 101, Triggers Introduction** for trigger configuration steps.

Examples To enable trigger 172, so that it can activate when its trigger conditions are met, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 172
awplus(config-trigger)# active
```

To disable trigger 182, preventing it from activating when its trigger conditions are met, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 182
awplus(config-trigger)# no active
```

Related Commands **show trigger**
trigger

day

This command specifies the days or date that the can trigger activate on. You can specify either:

- A specific date
- A specific day of the week
- A list of days of the week
- every day

By default, the trigger can activate on any day.

Syntax

```
day every-day
day <1-31> <month> <2000-2035>
day <weekday>
```

Parameter	Description
`every-day`	Sets the trigger so that it can activate on any day.
<1-31>	Day of the month the trigger is permitted to activate on.
<month>	Sets the month that the trigger is permitted to activate on. Valid keywords are: **january, february, march, april, may, june, july, august, september, october, november**, and **december**.
<2000-2035>	Sets the year that the trigger is permitted to activate in.
<weekday>	Sets the days of the week that the trigger can activate on. You can specify one or more week days in a space separated list. Valid keywords are: **monday, tuesday, wednesday, thursday, friday, saturday**, and **sunday**.

Mode Trigger Configuration

Usage For example trigger configurations that use the **day** command, see **"Restrict Internet Access" on page 102.2** and **"Turn Off Power to Port LEDs" on page 102.7**.

Examples To permit trigger 55 to activate on the `1 Jun 2010`, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 55
awplus(config-trigger)# day 1 Jun 2010
```

To permit trigger 12 to activate on a Mondays, Wednesdays and Fridays, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 12
awplus(config-trigger)# day monday wednesday friday
```

Related Commands **show trigger**
trigger

debug trigger

This command enables trigger debugging. This generates detailed messages about how your device is processing the trigger commands and activating the triggers.

The **no** variant of this command disables trigger debugging.

Syntax

```
debug trigger
no debug trigger
```

Mode Privilege Exec

Examples To start trigger debugging, use the command:

```
awplus# debug trigger
```

To stop trigger debugging, use the command:

```
awplus# no trigger
```

Related Commands

show debugging trigger
show trigger
test
trigger
undebug trigger

description (trigger)

This command adds an optional description to help you identify the trigger. This description is displayed in show command outputs and log messages.

The **no** variant of this command removes a trigger's description. The show command outputs and log messages stop displaying a description for this trigger.

Syntax

```
description <description>
no description
```

Parameter	Description
<description>	A word or phrase that uniquely identifies this trigger or its purpose. Valid characters are any printable character and spaces, up to a maximum of 40 characters.

Mode Trigger Configuration

Examples To give trigger 240 the description `daily status report`, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 240
awplus(config-trigger)# description daily status report
```

To remove the description from trigger `36`, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 36
awplus(config-trigger)# no description
```

Related Commands
show trigger
test
trigger

repeat

This command specifies the number of times that a trigger is permitted to activate. This allows you to specify whether you want the trigger to activate:

- only the first time that the trigger conditions are met
- a limited number of times that the trigger conditions are met
- an unlimited number of times

Once the trigger has reached the limit set with this command, the trigger remains in your configuration but cannot be activated. Use the **repeat** command again to reset the trigger so that it is activated when its trigger conditions are met.

By default, triggers can activate an unlimited number of times. To reset a trigger to this default, specify either **yes** or **forever**.

Syntax `repeat {forever|no|once|yes|<1-4294967294>}`

Parameter	Description
`yes|forever`	The trigger repeats indefinitely, or until disabled.
`no|once`	The trigger activates only once.
<1-4292967294>	The trigger repeats the set number of times.

Mode Trigger Configuration

Examples To allow trigger `21` to activate only once, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 21
awplus(config-trigger)# repeat no
```

To allow trigger `22` to activate an unlimited number of times whenever its trigger conditions are met, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 22
awplus(config-trigger)# repeat forever
```

To allow trigger `23` to activate only the first `10` times the conditions are met, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 23
awplus(config-trigger)# repeat 10
```

Related Commands **show trigger**
trigger

script

This command specifies one or more scripts that are to be run when the trigger activates. You can add up to five scripts to a single trigger.

The sequence in which the trigger runs the scripts is specified by the number you set before the name of the script file. One script is executed completely before the next script begins.

Scripts may be either ASH shell scripts, indicated by a **.sh** filename extension suffix, or AlliedWare Plus™ scripts, indicated by a **.scp** filename extension suffix. AlliedWare Plus™ scripts only need to be readable.

The **no** variant of this command removes one or more scripts from the trigger's script list. The scripts are identified by either their name, or by specifying their position in the script list. The **all** parameter removes all scripts from the trigger.

Syntax

```
script <1-5> {<filename>}
no script {<1-5>|<filename>|all}
```

Parameter	Description
<1-5>	The position of the script in execution sequence. The trigger runs the lowest numbered script first.
<filename>	The path to the script file.

Mode Trigger Configuration

Examples To configure trigger 71 to run the script flash:/cpu_trig.sh in position 3 when the trigger activates, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 71
awplus(config-trigger)# script 3 flash:/cpu_trig.sh
```

To configure trigger 99 to run the scripts **flash:reconfig.scp**, **flash:cpu_trig.sh** and **flash:email.scp** in positions 2, 3 and 5 when the trigger activates, use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 99
awplus(config-trigger)# script 2 flash:/reconfig.scp 3 flash:/
cpu_trig.sh 5 flash:/email.scp
```

To remove the scripts 1, 3 and 4 from trigger 71's script list, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 71
awplus(config-trigger)# no script 1 3 4
```

To remove the script flash:/cpu_trig.sh from trigger 71's script list, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 71
awplus(config-trigger)# no script flash:/cpu_trig.sh
```

To remove all the scripts from trigger 71's script list, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 71
awplus(config-trigger)# no script all
```

Related Commands

show trigger
trigger

show debugging trigger

This command displays the current status for trigger utility debugging. Use this command to show when trigger debugging has been turned on or off from the **debug trigger** command.

Syntax `show debugging trigger`

Mode User Exec and Privileged Exec

Example To display the current configuration of trigger debugging, use the command:

```
awplus# show debugging trigger
```

Output **Figure 103-1: Example output from the show debugging trigger command**

```
awplus#debug trigger
awplus#show debugging trigger
Trigger debugging status:
  Trigger debugging is on

awplus#no debug trigger
awplus#show debugging trigger
Trigger debugging status:
  Trigger debugging is off
```

Related Commands **debug trigger**

show running-config trigger

This command displays the current running configuration of the trigger utility.

Syntax `show running-config trigger`

Mode Privileged Exec

Example To display the current configuration of the trigger utility, use the command:

```
awplus# show running-config trigger
```

Related Commands **show trigger**

show trigger

This command displays configuration and diagnostic information about the triggers configured on the device. Specify the **show trigger** command without any options to display a summary of the configuration of all triggers.

Syntax `show trigger [<1-250>|counter|full]`

Parameter	Description
<1-250>	Displays detailed information about a specific trigger, identified by its trigger ID.
`counter`	Displays statistical information about all triggers.
`full`	Displays detailed information about all triggers.

Mode Privileged Exec

Example To get summary information about all triggers, use the following command:

```
awplus# show trigger
```

Figure 103-2: Example output from the show trigger command

```
awplus#show trigger
TR# Type & Details       Name                 Ac Te Tr Repeat     #Scr Days/Date
-------------------------------------------------------------------------------
003 CPU (80% any)        Busy CPU             Y  N  Y  5           1   smtwtfs
005 Periodic (30 min)    Regular status check Y  N  N  Continuous  1   -mtwtf-
007 Memory (85% up)      High mem usage       Y  N  Y  8           1   smtwtfs
011 Time (00:01)         Weekend access       Y  N  Y  Continuous  1   ------s
013 Reboot                                    Y  N  Y  Continuous  2   smtwtfs
017 Interface (vlan1 ... Change config for... Y  N  Y  Once        1   2-apr-2008
019 Ping-poll (5 up)     Connection to svr1   Y  N  Y  Continuous  1   smtwtfs
-------------------------------------------------------------------------------
```

Table 103-1: Parameters in the output of the show trigger command

Parameter	Description
`TR#`	Trigger identifier (ID).
`Type & Details`	The trigger type, followed by the trigger details in brackets.
`Name`	Descriptive name of the trigger configured with the **description (trigger)** command.
`Ac`	Whether the trigger is active (Y), or inactive (N).
`Te`	Whether the trigger is in test mode (Y) or not (N).
`Tr`	Whether or not the trigger is enabled to send SNMP traps. See the **trap** command.
`Repeat`	Whether the trigger repeats continuously, and if not, the configured repeat count for the trigger. To see the number of times a trigger has activated, use the **show trigger <1-250>** command.

Table 103-1: Parameters in the output of the show trigger command(cont.)

Parameter	Description
`#Scr`	Number of scripts associated with the trigger.
`Days/Date`	Days or date when the trigger may be activated. For the days options, the days are shown as a seven character string representing Sunday to Saturday. A hyphen indicates days when the trigger cannot be activated.

To display detailed information about trigger 3, use the command:

```
awplus# show trigger 3
```

Figure 103-3: Example output from the show trigger command for a specific trigger

```
awplus#show trigger 3
Trigger Configuration Details
-------------------------------------------------------------
Trigger ..................... 1
Description ................. display cpu usage when pass 80%
Type and details ............ CPU (80% up)
Days ........................ 26-nov-2007
After ....................... 00:00:00
Before ...................... 23:59:59
Active ...................... Yes
Test ........................ No
Trap ........................ Yes
Repeat ...................... 123 (0)
Modified .................... Tue Dec 20 02:26:03 1977
Number of activations ....... 0
Last activation ............. not activated
Number of scripts ........... 1
  1. shocpu.scp
  2. <not configured>
  3. <not configured>
  4. <not configured>
  5. <not configured>
-------------------------------------------------------------
```

To display detailed information about all triggers, use the command:

```
awplus# show trigger full
```

Figure 103-4: Example output from the show trigger full command

```
awplus#show trigger full
Trigger Configuration Details
-------------------------------------------------------------
 Trigger ..................... 1
 Description ................. <no description>
 Days ........................ smtwtfs
 After ....................... 00:00:00
 Before ...................... 23:59:59
 Active ...................... Yes
 Test ........................ No
 Trap ........................ Yes
 Repeat ...................... Continuous
 Modified .................... Fri Sep  3 14:45:56 2010
 Number of activations ....... 0
 Last activation ............. not activated
 Number of scripts ........... 0
   1. <not configured>
   2. <not configured>
   3. <not configured>
   4. <not configured>
   5. <not configured>

 Trigger ..................... 2
 Description ................. <no description>
 Days ........................ smtwtfs
 After ....................... 00:00:00
 Before ...................... 23:59:59
 Active ...................... Yes
 Test ........................ No
 Trap ........................ Yes
 Repeat ...................... Continuous
 Modified .................... Fri Sep  3 14:45:56 2010
 Number of activations ....... 0
 Last activation ............. not activated
 Number of scripts ........... 0
   1. <not configured>
   2. <not configured>
   3. <not configured>
   4. <not configured>
   5. <not configured>

Trigger ..................... 3
Description ................. Busy CPU
Type and details ............ CPU (80% up)
Days ........................ smtwtfs
Active ...................... Yes
Test ........................ No
Trap ........................ Yes
Repeat ...................... Continuous
Modified .................... Fri Feb 2 17:05:16 2007
Number of activations ....... 0
Last activation ............. not activated
Number of scripts ........... 2
  1. flash:/cpu_alert.sh
  2. flash:/reconfig.scp
  3. <not configured>
  4. <not configured>
  5. <not configured>
-------------------------------------------------------------
```

Table 103-2: Parameters in the output of the show trigger full and show trigger commands for a specific trigger

Parameter	Description
`Trigger`	The ID of the trigger.
`Description`	Descriptive name of the trigger.
`Type and details`	The trigger type and its activation conditions.
`Days`	The days on which the trigger is permitted to activate.
`Date`	The date on which the trigger is permitted to activate. Only displayed if configured, in which case it replaces "Days".
`Active`	Whether or not the trigger is permitted to activate.
`Test`	Whether or not the trigger is operating in diagnostic mode.
`Trap`	Whether or not the trigger is enabled to send SNMP traps.
`Repeat`	Whether the trigger repeats an unlimited number of times (Continuous) or for a set number of times. When the trigger can repeat only a set number of times, then the number of times the trigger has been activated is displayed in brackets.
`Modified`	The date and time of the last time that the trigger was modified.
`Number of activations`	Number of times the trigger has been activated since the last restart of the device.
`Last activation`	The date and time of the last time that the trigger was activated.
`Number of scripts`	How many scripts are associated with the trigger, followed by the names of the script files in the order in which they run.

To display counter information about all triggers use the command:

```
awplus# show trigger counter
```

Figure 103-5: Example output from the show trigger counter command

```
awplus#show trigger counter
Trigger Module Counters
-------------------------------------------------------
Trigger activations ........................... 0
Time triggers activated today ................. 0
Periodic triggers activated today ............. 0
Interface triggers activated today ............ 0
Resource triggers activated today ............. 0
Reboot triggers activated today ............... 0
Ping-poll triggers activated today ............ 0
-------------------------------------------------------
```

Table 103-3: Parameters in the output of the show trigger counter command

Parameter	Description
`Trigger activations`	Number of times a trigger has been activated.
`Time triggers activated today`	Number of times a time trigger has been activated today.
`Periodic triggers activated today`	Number of times a periodic trigger has been activated today.
`Interface triggers activated today`	Number of times an interface trigger has been activated today.
`Resource triggers activated today`	Number of times a CPU or memory resource trigger has been activated today.
`Ping-poll triggers activated today`	Number of times a ping-poll trigger has been activated today.

Related Commands **trigger**

test

This command puts the trigger into a diagnostic mode. In this mode the trigger may activate but when it does it will not run any of the trigger's scripts. A log message will be generated to indicate when the trigger has been activated.

The **no** variant of this command takes the trigger out of diagnostic mode, restoring normal operation. When the trigger activates the scripts associated with the trigger will be run, as normal.

Syntax

```
test
no test
```

Mode Trigger Configuration

Usage Configure a trigger first before you use this command to diagnose it. See the **Configuring a Trigger** section in **Chapter 101, Triggers Introduction** for trigger configuration steps.

Examples To put trigger 5 into diagnostic mode, where no scripts will be run when the trigger activates, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 5
awplus(config-trigger)# test
```

To take trigger 205 out of diagnostic mode, restoring normal operation, use the commands:

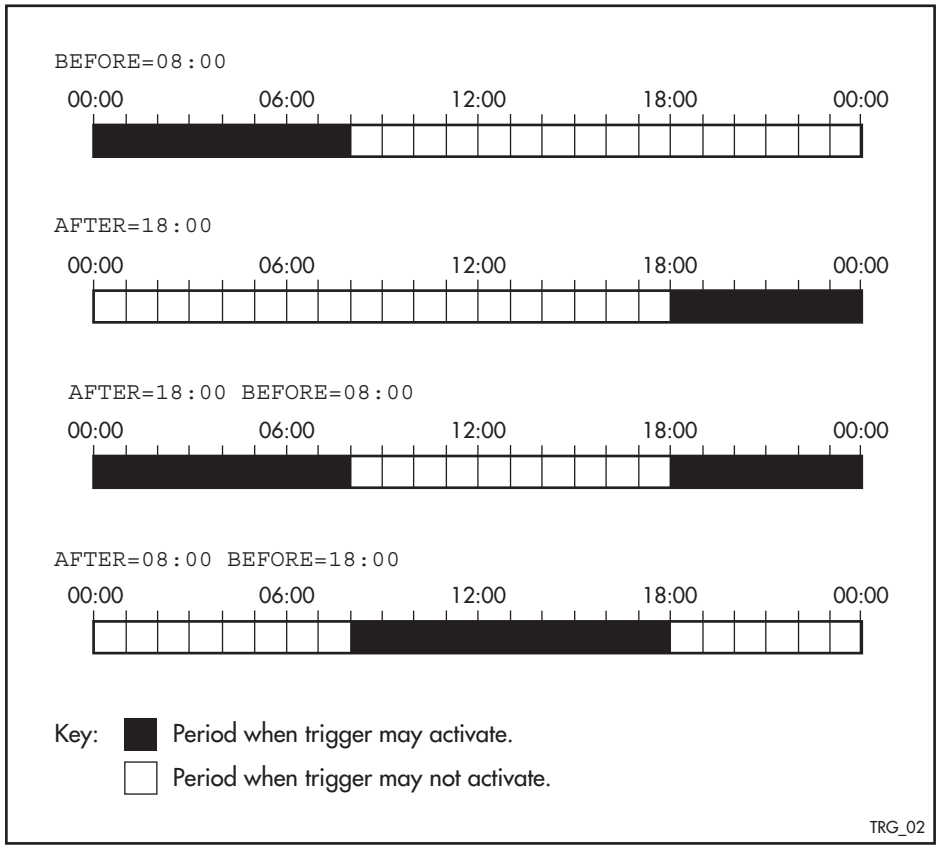
```
awplus# configure terminal
awplus(config)# trigger 205
awplus(config-trigger)# no test
```

Related Commands **show trigger**
trigger

time (trigger)

This command specifies the time of day when the trigger is permitted to activate. The after parameter specifies the start of a time period that extends to midnight during which trigger may activate. By default the value of this parameter is 00:00:00 (am); that is, the trigger may activate at any time. The **before** parameter specifies the end of a time period beginning at midnight during which the trigger may activate. By default the value of this parameter is 23:59:59; that is, the trigger may activate at any time. If the value specified for **before** is later than the value specified for **after**, a time period from "after" to "before" is defined, during which the trigger may activate. This command is not applicable to time triggers (**type time**).

The following figure illustrates how the **before** and **after** parameters operate.



Syntax `time {[after <hh:mm:ss>] [before <hh:mm:ss>]}`

Parameter	Description
`after <hh:mm:ss>`	The earliest time of day when the trigger may be activated.
`before <hh:mm:ss>`	The latest time of day when the trigger may be activated.

Mode Trigger Configuration

Usage For example trigger configurations that use the **time (trigger)** command, see **"Restrict Internet Access" on page 102.2** and **"Turn Off Power to Port LEDs" on page 102.7**.

Examples To allow trigger `63` to activate between midnight and 10:30am, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 63
awplus(config-trigger)# time before 10:30:00
```

To allow trigger `64` to activate between 3:45pm and midnight, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 64
awplus(config-trigger)# time after 15:45:00
```

To allow trigger `65` to activate between 10:30am and 8:15pm, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 65
awplus(config-trigger)# time after 10:30:00 before 20:15:00
```

Related Commands **show trigger**
trigger

trap

This command enables the specified trigger to send SNMP traps.

Use the **no** variant of this command to disable the sending of SNMP traps from the specified trigger.

Syntax

```
trap
no trap
```

Default SNMP traps are enabled by default for all defined triggers.

Mode Trigger Configuration

Usage You must configure SNMP before using traps with triggers. See the following SNMP chapters:

Chapter 93, SNMP Introduction

Chapter 94, SNMP Commands

Chapter 95, SNMP MIBs

Since SNMP traps are enabled by default for all defined triggers, a common usage will be for the **no** variant of this command to disable SNMP traps from a specified trap if the trap is only periodic. Refer in particular to **AT-TRIGGER-MIB** for further information about the relevant SNMP MIB.

Examples To enable SNMP traps to be sent from trigger 5, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 5
awplus(config-trigger)# trap
```

To disable SNMP traps being sent from trigger 205, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 205
awplus(config-trigger)# no trap
```

Related Commands

trigger
show trigger

trigger

This command is used to access the Trigger Configuration mode for the specified trigger. Once Trigger Configuration mode has been entered the trigger type information can be configured and the trigger scripts and other operational parameters can be specified. At a minimum the trigger type information must be specified before the trigger can become active.

The **no** variant of this command removes a specified trigger and all configuration associated with it.

Syntax

```
trigger <1-250>
no trigger <1-250>
```

Parameter	Description
<1-250>	A trigger ID.

Mode Global Configuration

Examples To enter trigger configuration mode for trigger `12` use the command:

```
awplus# trigger 12
```

To completely remove all configuration associated with trigger `12`, use the command:

```
awplus# no trigger 12
```

Related Commands

show trigger
trigger activate

trigger activate

This command is used to manually activate a specified trigger from the Privileged Exec mode, which has been configured with the **trigger** command from the Global Configuration mode.

Syntax `trigger activate <1-250>`

Parameter	Description
<1-250>	A trigger ID.

Mode Privileged Exec

Usage This command manually activates a trigger without the normal trigger conditions being met.

The trigger is activated even if it is configured as inactive. The scripts associated with the trigger will be executed even if the trigger is in the diagnostic test mode.

Triggers activated manually do not have their repeat counts decremented or their 'last triggered' time updated, and do not result in updates to the '[type] triggers today' counters.

Example To manually activate trigger `12` use the command:

```
awplus# trigger activate 12
```

Related Commands **show trigger**
trigger

type atmf node

This command configures a trigger to be activated at an AMF node join event or leave event.

Syntax `type atmf node {join|leave}`

Parameter	Description
`join`	AMF node join event.
`leave`	AMF node leave event.

Mode Trigger Configuration

Caution **Only configure this trigger on one device because it is a network wide event.**

Example 1 To configure trigger 5 to activate at an AMF node leave event, use the following commands. In this example the command is entered on node-1:

```
node1(config)# trigger 5
node1(config-trigger) type atmf node leave
```

Example 2 The following commands will configure trigger 5 to activate if an AMF node join event occurs on any node within the working set:

```
node1# atmf working-set group all
```

This command returns the following display:

```
====================
node1, node2, node3:
====================

Working set join
```

Note that the running the above command changes the prompt from the name of the local node, to the name of the AMF-Network followed, in square brackets, by the number of member nodes in the working set.

```
AMF-Net[3]# conf t
AMF-Net[3](config)# trigger 5
AMF-Net[3](config-trigger)# type atmf node leave
AMF-Net[3](config-trigger)# description "E-mail on AMF Exit"
AMF-Net[3](config-trigger)# active
```

Enter the name of the script to run at the trigger event.

```
AMF-Net[3](config-trigger)# script 1 email_me.scp
AMF-Net[3](config-trigger)# end
```

Display the trigger configurations

```
AMF-Net[3]# show trigger
```

This command returns the following display:

```
=======
node1:
=======

TR# Type & Details     Description          Ac Te Tr Repeat     #Scr Days/Date
-------------------------------------------------------------------------------
001 Periodic (2 min)   Periodic Status Chk  Y  N  Y Continuous  1   smtwtfs
005 ATMF node (leave)  E-mail on ATMF Exit  Y  N  Y Continuous  1   smtwtfs
-------------------------------------------------------------------------------


==============
Node2, Node3,
==============

TR# Type & Details     Description          Ac Te Tr Repeat     #Scr Days/Date
-------------------------------------------------------------------------------
005 ATMF node (leave)  E-mail on ATMF Exit  Y  N  Y Continuous  1   smtwtfs
-------------------------------------------------------------------------------
```

Display the triggers configured on each of the nodes in the AMF Network.

```
AMF-Net[3]# show running-config trigger
```

This command returns the following display:

```
========
Node1:
========

trigger 1
 type periodic 2
 script 1 atmf.scp
trigger 5
 type atmf node leave
description "E-mail on ATMF Exit"
 script 1 email_me.scp
!


============
Node2, Node3:
============

trigger 5
 type atmf node leave
description "E-mail on ATMF Exit"
 script 1 email_me.scp
!
```

Related Commands **show trigger**

type cpu

This command configures a trigger to activate based on CPU usage level. Selecting the **up** option causes the trigger to activate when the CPU usage exceeds the specified usage level. Selecting the **down** option causes the trigger to activate when CPU usage drops below the specified usage level. Selecting **any** causes the trigger to activate in both situations. The default is **any**.

Syntax `type cpu <1-100> [up|down|any]`

Parameter	Description
`<1-100>`	The percentage of CPU usage at which to trigger.
`up`	Activate when CPU usage exceeds the specified level.
`down`	Activate when CPU usage drops below the specified level
`any`	Activate when CPU usage passes the specified level in either direction

Mode Trigger Configuration

Usage For an example trigger configuration that uses the **type cpu** command, see **"Capture Unusual CPU and RAM Activity" on page 102.4**.

Examples To configure trigger `28` to be a CPU trigger that activates when CPU usage exceeds 80% use the following commands:

```
                awplus# configure terminal
        awplus(config)# trigger 28
awplus(config-trigger)# type cpu 80 up
```

To configure trigger `5` to be a CPU trigger that activates when CPU usage either rises above or drops below 65%, use the following commands:

```
                awplus# configure terminal
        awplus(config)# trigger 5
awplus(config-trigger)# type cpu 65
```

or

```
                awplus# configure terminal
        awplus(config)# trigger 5
awplus(config-trigger)# type cpu 65 any
```

Related Commands **show trigger**
trigger

type interface

This command configures a trigger to activate based on the link status of an interface. The trigger can be activated when the interface becomes operational by using the **up** option, or when the interface closes by using the **down** option. The trigger can also be configured to activate when either one of these events occurs by using the **any** option.

Syntax `type interface <interface> [up|down|any]`

Parameter	Description
`<interface>`	Interface name. This can be the name of a switch port, an eth-management port, or a VLAN.
`up`	Activate when interface becomes operational.
`down`	Activate when the interface closes.
`any`	Activate when any interface link status event occurs.

Mode Trigger Configuration

Example To configure trigger `19` to be an interface trigger that activates when `port1.0.2` becomes operational, use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 19
awplus(config-trigger)# type interface port1.0.2 up
```

Related Commands **show trigger**
trigger

type memory

This command configures a trigger to activate based on RAM usage level. Selecting the **up** option causes the trigger to activate when memory usage exceeds the specified level. Selecting the **down** option causes the trigger to activate when memory usage drops below the specified level. Selecting **any** causes the trigger to activate in both situations. The default is **any**.

Syntax `type memory <1-100> [up|down|any]`

Parameter	Description
<1-100>	The percentage of memory usage at which to trigger.
`up`	Activate when memory usage exceeds the specified level.
`down`	Activate when memory usage drops below the specified level.
`any`	Activate when memory usage passes the specified level in either direction.

Mode Trigger Configuration

Examples To configure trigger `12` to be a memory trigger that activates when memory usage exceeds 50% use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 12
awplus(config-trigger)# type memory 50 up
```

To configure trigger `40` to be a memory trigger that activates when memory usage either rises above or drops below 65%, use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 40
awplus(config-trigger)# type memory 65
```

or

```
awplus# configure terminal
awplus(config)# trigger 40
awplus(config-trigger)# type memory 65 any
```

Related Commands **show trigger**
trigger

type periodic

This command configures a trigger to be activated at regular intervals. The time period between activations is specified in minutes.

Syntax `type periodic <1-1440>`

Parameter	Description
<1-1440>	The number of minutes between activations.

Mode Trigger Configuration

Usage A combined limit of 10 triggers of the type periodic and time can be configured. If you attempt to add more than 10 triggers the following error message is displayed:

```
% Cannot configure more than 10 triggers with the type time or
periodic
```

For an example trigger configuration that uses the **type periodic** command, see **"See Daily Statistics" on page 102.6**.

Example To configure trigger `44` to activate periodically at `10` minute intervals use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 44
awplus(config-trigger)# type periodic 10
```

Related Commands **show trigger**
trigger

type ping-poll

This command configures a trigger that activates when Ping Polling identifies that a target device's status has changed. This allows you to run a configuration script when a device becomes reachable or unreachable.

Syntax `type ping-poll <1-100> {up|down}`

Parameter	Description
<1-100>	The ping poll ID.
`up`	The trigger activates when ping polling detects that the target is reachable.
`down`	The trigger activates when ping polling detects that the target is unreachable.

Mode Trigger Configuration

Example To configure trigger `106` to activate when ping poll `12` detects that its target device is now unreachable, use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 106
awplus(config-trigger)# type ping-poll 12 down
```

Related Commands **show trigger**
trigger

type reboot

This command configures a trigger that activates when your device is rebooted.

Syntax `type reboot`

Mode Trigger Configuration

Example To configure trigger `32` to activate when your device reboots, use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 32
awplus(config-trigger)# type reboot
```

Related Commands **show trigger**
trigger

type stack disabled-master

This command (configured to the stack) configures a trigger to activate on a stack member if it becomes the disabled master.

A disabled master has the same configuration as the active master, but has all its links shutdown.

Although this command could activate any trigger script, the intention here is that the script will reactivate the links from their previously shutdown state, to enable the user to manage the switch. An appropriate trigger script must already exist that will apply the **no shutdown** command on page 14.15 on the deactivated links.

Caution **It is important that any ports that are configured as trunked ports across master and stack members are disabled at their stack member termination when operating in the fallback configuration. Otherwise, the trunked ports will not function correctly on the switch that is connected downstream.**

If the **stack virtual-mac** command on page 111.32 command is enabled, the stack uses a virtual MAC address. The stack will always use this MAC address and the new elected master will still retain the originally configured virtual MAC address. If the **stack virtual-mac** command is disabled, the stack will use the MAC address of the current master. If the stack master fails, the stack MAC address changes to reflect the new master's MAC address. See **"Fixed or Virtual MAC Addressing" on page 110.12** for information on virtual MAC addresses.

Syntax

```
type stack disabled-master
```

Mode Trigger Configuration

Examples To configure trigger `82` to activate on a device if it becomes the disabled master, use the commands:

`awplus#`	
`configure terminal`	Enter the Global Configuration mode
`awplus(config)#`	
`trigger 82`	Enter the Trigger Configuration mode for trigger 82
`awplus(config-trigger)#`	
`type stack disabled master`	Sets the type of trigger
`awplus(config-trigger)#`	
`script 1 flash:/disabled.scp`	
`awplus(config-trigger)#`	
`exit`	

Related Commands

stack disabled-master-monitoring
trigger
type stack master-fail
type stack member
type stack link

type stack link

This command (configured to the stack) initiates the action of a pre-configured trigger to occur when a stacking link is either activated or deactivated.

Syntax `type stack link {up|down}`

Parameter	Description
`up`	Stack link up event
`down`	Stack link down event

Mode Trigger Configuration

Example To configure trigger `86` to activate when the stack link down event occurs, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 86
awplus(config-trigger)# type stack link down
```

Related Commands
show trigger
trigger
type stack master-fail

type stack master-fail

This command (configured to the stack) initiates the action of a pre-configured trigger to occur when the stack enters the fail-over state.

Syntax `type stack master-fail`

Mode Trigger Configuration

Example To configure trigger `86` to activate when stack master fail-over event occurs, use the commands:

```
awplus# configure terminal
awplus(config)# trigger 86
awplus(config-trigger)# type stack master-fail
```

Related Commands
stack disabled-master-monitoring
trigger
type stack disabled-master
type stack member
type stack link

type stack member

This command (configured to the stack) initiates the action of a pre-configured trigger to occur when a switch either joins or leaves the stack.

Syntax

```
type stack member {join|leave}
```

Parameter	Description
`join`	Neighbor join event
`leave`	Neighbor leave event

Mode Trigger Configuration

Example To configure a pre-configured trigger number `86` to activate when a new switch joins the stack.

Note that the number `86` has no particular significance: you can assign any (previously created) numbered trigger.

```
awplus# configure terminal
awplus(config)# trigger 86
awplus(config-trigger)# type stack member join
```

Related Commands

trigger
type stack master-fail
type stack link

type time

This command configures a trigger that activates at a specified time of day.

Syntax `type time <hh:mm>`

Parameter	Description
<hh:mm>	The time to activate the trigger.

Mode Trigger Configuration

Usage A combined limit of 10 triggers of the type time and type periodic can be configured. If you attempt to add more than 10 triggers the following error message is displayed:

```
% Cannot configure more than 10 triggers with the type time or
periodic
```

Example To configure trigger `86` to activate at `15:53`, use the following commands:

```
awplus# configure terminal
awplus(config)# trigger 86
awplus(config-trigger)# type time 15:53
```

Related Commands **show trigger**
trigger

undebug trigger

This command applies the functionality of the **no debug trigger** command.

Chapter 104: Ping Polling Introduction and Configuration

Introduction 104.2
How Ping Polling Works 104.2
Configuring Ping Polling 104.4
Creating a Polling Instance 104.4
Customizing a Polling Instance 104.5
Troubleshooting Ping Polling 104.6
Interaction with Other Protocols 104.7

Introduction

Ping polling lets your device regularly check whether it can reach other hosts on a network. It works by sending ICMP Echo Requests to a host and waiting for replies sent back. If ping polling indicates that a host's status has changed, then your device can respond to the new status. When a host is unreachable, ping polling continues monitoring the host's reachability.

You can configure triggers to activate when ping polling determines that the host's status has changed. For example, you could configure a trigger to run a script that opens and configures an alternative link if the host at the other end of a preferred link becomes unavailable. You could then configure a second trigger to run a script that automatically returns traffic to the preferred link as soon as it is available again.

How Ping Polling Works

To determine a host's reachability, your device regularly sends ICMP Echo Request packets ("pings") to the host. As long as your device receives ping responses from the host, it considers the host to be reachable. If your device does not receive a reply to a set number of ICMP Echo Requests, it considers that the host is unreachable. It continues to try to ping the device, at an increased rate. After it receives a set number of responses, it considers the device to be reachable again.

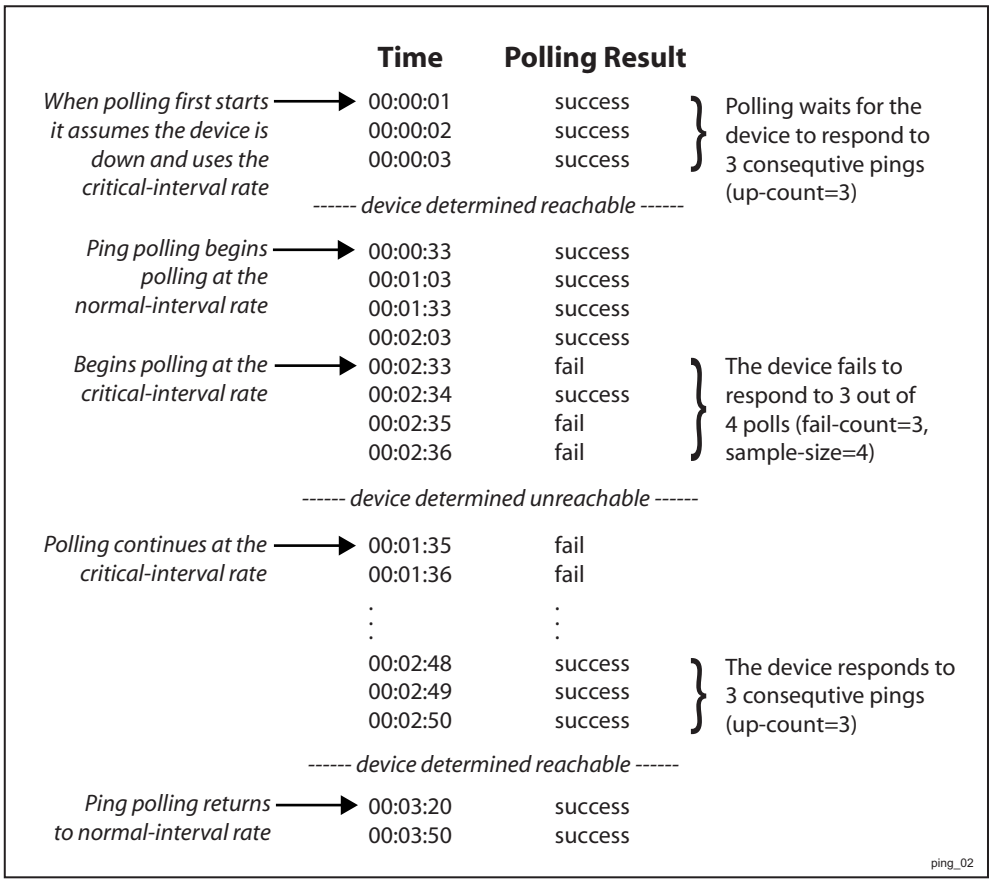
By default, a polling instance sends a ping every 30 seconds as long as it is receiving replies. The frequency of this polling is controlled by the normal-interval command. When a reply is not received, the polling instance increases the frequency at which it polls the device. This frequency is controlled by the critical-interval command, and by default, is set to send a packet every one second. It maintains this higher rate of polling until it has received sufficient consecutive replies.

The polling instance determines whether a device is reachable or unreachable based on the settings of the fail-count, sample-size, and up-count commands. To determine whether a device is reachable, the polling instance counts the number of failed pings within a set sample size. The sample size is set by the sample-size command, and by default is 5 ping responses. Within the sample size, the number of failed pings that means that the device is down is set by the fail-count command. By default this is set to 5. Once a polling instance has determined that a device is unreachable, it must receive a set number of consecutive replies before it changes the device's status back to reachable. This number is configured with the up-count command.

The following figure illustrates a polling instance where the device becomes unreachable, then reachable. It uses this configuration:

```
awplus(config-ping-poll)# fail-count 4
awplus(config-ping-poll)# sample-size 5
awplus(config-ping-poll)# up-count 3
awplus(config-ping-poll)# critical-interval 1
awplus(config-ping-poll)# normal-interval 30
```

Figure 104-1: Interaction between states and parameters for ping polling



On some operating systems, some servers may respond to a ping even if no other functionality is available, and therefore remain in an Up state while malfunctioning.

Responding to status changes

To configuring your device to determine and respond to changes in a device's reachability, you will need to:

- create a polling instance to periodically ping the device
- create scripts to run when the device becomes unreachable and when it becomes reachable again
- configure triggers to run these scripts

To set a trigger to activate when a device's status changes, its trigger type must be **ping-poll**. This is with the following command in the trigger's configuration mode:

```
awplus(config-trigger)# type ping-poll <1-100> {up|down}
```

where **up** activates the trigger when the device is reachable, and **down** activates the trigger when the device is unreachable.

If you use triggers to open a backup link to a remote device in the event of the primary link failing (rather than the remote device failing), the backup link and primary link must point to different IP addresses on the remote device. Otherwise, when the backup link points to the IP address that your device is polling, your device receives ping replies through the backup link, considers the device to be reachable again, and attempts to reopen the primary link instead of using the backup link. See **Chapter 101, Triggers Introduction** for more information about configuring Triggers with Ping Polling.

Configuring Ping Polling

This section contains:

- **Creating a Polling Instance**
  This explains how to quickly create a polling instance using the ping polling defaults.
- **Customizing a Polling Instance**
  This explains how to customize a ping poll and explains the other ping poll commands.
- **Troubleshooting Ping Polling**
  This explains how to use the debugging and monitoring commands for ping polling.

Creating a Polling Instance

The Ping Polling feature in the AlliedWare Plus™ OS allows you to easily configure polling instances with a minimum of commands. To configure a ping poll suitable for most network situations:

1. Create a polling instance by using the command:

```
awplus(config)# ping-poll <1-100>
```

   The range **<1-100>** identifies the polling instance in the trigger commands and in other ping poll commands. Your device can poll up to 100 IP addresses at once.

2. Set the IP address of the device you are polling by using the command:

```
awplus(config-ping-poll)# ip {<ip-address>|<ipv6-address>}
```

3. Enable the polling instance by using the command:

```
awplus(config-ping-poll)# active
```

4. If desired, set an optional description to identify the polling instance, by using the command:

```
awplus(config-ping-poll)# description <description>
```

You do not need to configure any other commands for most networks, because convenient defaults exist for all other ping poll settings. The following table summarizes the default configuration created.

Command	Default
Critical-interval	1 second
Fail-count	5
Length	32 bytes
Normal-interval	30 seconds

Command(cont.)	Default(cont.)
Sample-size	5
Source-ip	The IP address of the interface from which the ping packets are transmitted
Time-out	1 second
Up-count	30

Customizing a Polling Instance

Once you've created a polling instance using the ping-poll and **ip (ping-polling)** command, you may wish to customize the polling instance for your network.

Packet size

If you find that larger packet types in your network are not reaching the polled device while smaller ones such as ping do, you can increase the data bytes included in the ping packets sent by the polling instance. This encourages the polling instance to change the device's status to unreachable when packet of the size you are interested in are being dropped. To change the number of bytes sent in the data portion of the ping packets, use the command:

```
awplus(config-ping-poll)# length <4-1500>
```

Response timeout

The polling instance determines that a device hasn't responded to a ping if one second elapses without a response to the ping. In networks where ping packets have a low priority, you may need to set the allowed response time to a longer time period. To change this, use the command:

```
awplus(config-ping-poll)# timeout <1-30>
```

Polling frequency

By default, a polling instance polls a reachable device every 30 seconds. You can change this by using the command:

```
awplus(config-ping-poll)# normal-interval <1-65536>
```

Once the polling instance has determined that a ping has failed, it starts polling the device at the frequency set as the critical interval—by default, one second. To change the frequency set by the critical interval, use the command:

```
awplus(config-ping-poll)# critical-interval <1-65536>
```

The critical interval enables the polling instance to quickly observe changes in the state of the device, and should be set to a much lower value than the normal interval.

Configuring when the device's status changes

The number of pings that the polling instance examines to consider a change in state is controlled by the interaction of the sample-size, fail-count, and up-count commands. See **"How Ping Polling Works" on page 104.2** for an example showing this interaction.

To determine whether a device is reachable, the polling instance counts the number of failed pings within a sample of a set size. The sample size is 5 pings by default. To change the sample size, use the command:

```
awplus(config-ping-poll)# sample-size <1-100>
```

To change the number of failed pings that the sample must have, use the command:

```
awplus(config-ping-poll)# fail-count <1-100>
```

If the sample size and fail count are the same, the unanswered pings must be consecutive. If the sample size is greater than the fail count, a device that does not always reply to pings may be declared unreachable.

The upcount is the number of consecutive pings that must be answered for the polling instance to consider the device reachable again. To change this from the default of 30, use the command:

```
awplus(config-ping-poll)# up-count <1-100>
```

Checking the configuration

To check the settings and status of the polling instance, use the command:

```
awplus(config-ping-poll)# show ping-poll [<1-100>|state {up|
                          down}] [brief]
```

Troubleshooting Ping Polling

To disable a polling instance, use the command:

```
awplus(config-ping-poll)# no active
```

The polling instance no longer sends ICMP echo requests to the polled device and the counters for this polling instance are reset.

To clear the counters and change the status of a device to unreachable, enter the Privileged Exec mode and use the command:

```
awplus# clear ping-poll {<1-100>|all}
```

The polling instance changes to the polling frequency specified with the critical-interval command. The device status changes to reachable once the device responses have reached the **up-count**.

To start debugging for ping polling, use the command:

```
awplus# debug ping-poll <1-100>
```

Interaction with Other Protocols

Ping polling does not work if the polled host, your device, or any intermediate routers or switches are configured to drop ICMP Echo Requests and Replies.

Ping and Traceroute

Ping and Traceroute are not affected by ping polling. You can enter ping and trace commands at any time and independent of the polling.

Chapter 105: Ping-Polling Commands

Command List 105.2
active (ping-polling) 105.3
clear ping-poll 105.4
critical-interval 105.5
debug ping-poll 105.6
description (ping-polling) 105.7
fail-count 105.8
ip (ping-polling) 105.9
length (ping-poll data) 105.10
normal-interval 105.11
ping-poll 105.12
sample-size 105.13
show counter ping-poll 105.14
show ping-poll 105.16
source-ip 105.20
timeout (ping polling) 105.21
up-count 105.22
undebug ping-poll 105.22

Command List

This chapter provides an alphabetical reference for commands used to configure Ping Polling. For more information, see **Chapter 104, Ping Polling Introduction and Configuration**.

For information about modifying or redirecting the output from **show** commands to a file, see **"Controlling "show" Command Output" on page 1.37**.

Table 105-1: The following table lists the default values when configuring a ping poll

Default	Value
Critical-interval	1 second
Description	No description
Fail-count	5
Length	32 bytes
Normal-interval	30 seconds
Sample-size	5
Source-ip	The IP address of the interface from which the ping packets are transmitted
Time-out	1 second
Up-count	30

active (ping-polling)

This command enables a ping-poll instance. The polling instance sends ICMP echo requests to the device with the IP address specified by the **ip (ping-polling)** command.

By default, polling instances are disabled. When a polling instance is enabled, it assumes that the device it is polling is unreachable.

The **no** variant of this command disables a ping-poll instance. The polling instance no longer sends ICMP echo requests to the polled device. This also resets all counters for this polling instance.

Syntax

```
active
no active
```

Mode Ping-Polling Configuration

Examples To activate the ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# active
```

To disable the ping-poll instance 43 and reset its counters, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no active
```

Related Commands

debug ping-poll
ip (ping-polling)
ping-poll
show ping-poll

clear ping-poll

This command resets the specified ping poll, or all ping poll instances. This clears the ping counters, and changes the status of polled devices to unreachable. The polling instance changes to the polling frequency specified with the **critical-interval** command. The device status changes to reachable once the device responses have reached the **up-count**.

Syntax `clear ping-poll {<1-100>|all}`

Parameter	Description
`<1-100>`	A ping poll ID number. The specified ping poll instance has its counters cleared, and the status of the device it polls is changed to unreachable.
`all`	Clears the counters and changes the device status of all polling instances.

Mode Privileged Exec

Examples To reset the ping poll instance 12, use the command:

```
awplus# clear ping-poll 12
```

To reset all ping poll instances, use the command:

```
awplus# clear ping-poll all
```

Related Commands
active (ping-polling)
ping-poll
show ping-poll

critical-interval

This command specifies the time period in seconds between pings when the polling instance has not received a reply to at least one ping, and when the device is unreachable.

This command enables the device to quickly observe changes in state, and should be set to a much lower value than the **normal-interval** command.

The **no** variant of this command sets the critical interval to the default of one second.

Syntax

```
critical-interval <1-65536>
no critical-interval
```

Parameter	Description
<1-65536>	Time in seconds between pings, when the device has failed to a ping, or the device is unreachable.

Default The default is 1 second.

Mode Ping-Polling Configuration

Examples To set the critical interval to 2 seconds for the ping-polling instance 99, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 99
awplus(config-ping-poll)# critical-interval 2
```

To reset the critical interval to the default of one second for the ping-polling instance 99, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 99
awplus(config-ping-poll)# no critical-interval
```

Related Commands

fail-count
normal-interval
sample-size
show ping-poll
timeout (ping polling)
up-count

debug ping-poll

This command enables ping poll debugging for the specified ping-poll instance. This generates detailed messages about ping execution.

The **no** variant of this command disables ping-poll debugging for the specified ping-poll.

Syntax

```
debug ping-poll <1-100>
no debug ping-poll {<1-100>|all}
```

Parameter	Description
<1-100>	A unique ping poll ID number.
`all`	Turn off all ping-poll debugging.

Mode Privileged Exec

Examples To enable debugging for ping-poll instance 88, use the command:

```
awplus# debug ping-poll 88
```

To disable all ping poll debugging, use the command:

```
awplus# no debug ping-poll all
```

To disable debugging for ping-poll instance 88, use the command:

```
awplus# no debug ping-poll 88
```

Related Commands

active (ping-polling)
clear ping-poll
ping-poll
show ping-poll
undebug ping-poll

description (ping-polling)

This command specifies a string to describe the ping-polling instance. This allows the ping-polling instance to be recognized easily in show commands. Setting this command is optional.

By default ping-poll instances do not have a description.

Use the **no** variant of this command to delete the description set.

Syntax

```
description <description>
no description
```

Parameter	Description
<description>	The description of the target. Valid characters are any printable character and spaces. There is no maximum character length.

Mode Ping-Polling Configuration

Examples To add the text "Primary Gateway" to describe the ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# description Primary Gateway
```

To delete the description set for the ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no description
```

Related Commands **ping-poll**
show ping-poll

fail-count

This command specifies the number of pings that must be unanswered, within the total number of pings specified by the **sample-size** command, for the ping-polling instance to consider the device unreachable.

If the number set by the **sample-size** command and the **fail-count** commands are the same, then the unanswered pings must be consecutive. If the number set by the **sample-size** command is greater than the number set by the **fail-count** command, then a device that does not always reply to pings may be declared unreachable.

The **no** variant of this command resets the fail count to the default.

Syntax

```
fail-count <1-100>
no fail-count
```

Parameter	Description
<1-100>	The number of pings within the sample size that a reachable device must fail to respond to before it is classified as unreachable.

Default The default is 5.

Mode Ping-Polling Configuration

Examples To specify the number of pings that must fail within the sample size to determine that a device is unreachable for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# fail-count 5
```

To reset the fail-count to its default of 5 for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no fail-count
```

Related Commands

critical-interval
normal-interval
ping-poll
sample-size
show ping-poll
timeout (ping polling)
up-count

ip (ping-polling)

This command specifies the IPv4 address of the device you are polling.

Syntax `ip {<ip-address>|<ipv6-address>}`

Parameter	Description
<ip-address>	An IPv4 address in dotted decimal notation `A.B.C.D`
<ipv6-address>	An IPv6 address in hexadecimal notation `X:X::X:X`

Mode Ping-Polling Configuration

Examples To set ping-poll instance 5 to poll the device with the IP address `192.168.0.1`, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 5
awplus(config-ping-poll)# ip 192.168.0.1
```

To set ping-poll instance 10 to poll the device with the IPv6 address `2001:db8::`, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 10
awplus(config-ping-poll)# ip 2001:db8::
```

Related Commands
ping-poll
source-ip
show ping-poll

length (ping-poll data)

This command specifies the number of data bytes to include in the data portion of the ping packet. This allows you to set the ping packets to a larger size if you find that larger packet types in your network are not reaching the polled device, while smaller packets are getting through. This encourages the polling instance to change the device's status to unreachable when the network is dropping packets of the size you are interested in.

The **no** variant of this command resets the data bytes to the default of 32 bytes.

Syntax

```
length <4-1500>
no length
```

Parameter	Description
<4-1500>	The number of data bytes to include in the data portion of the ping packet.

Default The default is 32.

Mode Ping-Polling Configuration

Examples To specify that ping-poll instance 12 sends ping packet with a data portion of 56 bytes, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 12
awplus(config-ping-poll)# length 56
```

To reset the number of data bytes in the ping packet to the default of 32 bytes for ping-poll instance 3, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 12
awplus(config-ping-poll)# length
```

Related Commands **ping-poll**
show ping-poll

normal-interval

This command specifies the time period between pings when the device is reachable.

The **no** variant of this command resets the time period to the default of 30 seconds.

Syntax

```
normal-interval <1-65536>
no normal-interval
```

Parameter	Description
<1-65536>	Time in seconds between pings when the target is reachable.

Default The default is 30 seconds.

Mode Ping-Polling Configuration

Examples To specify a time period of 60 seconds between pings when the device is reachable for ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# normal-interval 60
```

To reset the interval to the default of 30 seconds for ping-poll instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no normal-interval
```

Related Commands

critical-interval
fail-count
ping-poll
sample-size
show ping-poll
timeout (ping polling)
up-count

ping-poll

This command enters the ping-poll configuration mode. If a ping-poll exists with the specified number, then this command enters its configuration mode. If no-ping poll exists with the specified number, then this command creates a new ping poll with this ID number.

To configure a ping-poll, create a ping poll using this command, and use the **ip (ping-polling)** command to specify the device you want the polling instance to poll. It is not necessary to specify any further commands unless you want to change a command's default.

The **no** variant of this command deletes the specified ping poll.

Syntax

```
ping-poll <1-100>
no ping-poll <1-100>
```

Parameter	Description
<1-100>	A unique ping poll ID number.

Mode Global Configuration

Examples To create ping-poll instance 3 and enter ping-poll configuration mode, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 3
awplus(config-ping-poll)#
```

To delete ping-poll instance 3, use the commands:

```
awplus# configure terminal
awplus(config)# no ping-poll 3
```

Related Commands

active (ping-polling)
clear ping-poll
debug ping-poll
description (ping-polling)
ip (ping-polling)
length (ping-poll data)
show ping-poll
source-ip

sample-size

This command sets the total number of pings that the polling instance inspects when determining whether a device is unreachable. If the number of pings specified by the **fail-count** command go unanswered within the inspected sample, then the device is declared unreachable.

If the numbers set in this command and **fail-count** command are the same, the unanswered pings must be consecutive. If the number set by this command is greater than that set with the **fail-count** command, a device that does not always reply to pings may be declared unreachable.

You cannot set this command's value lower than the **fail-count** value.

The polling instance uses the number of pings specified by the **up-count** command to determine when a device is reachable.

The **no** variant of this command resets this command to the default.

Syntax

```
sample-size <1-100>
no sample size
```

Parameter	Description
<1-100>	Number of pings that determines critical and up counts.

Default The default is 5.

Mode Ping-Polling Configuration

Examples To set the sample-size to 50 for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# sample-size 50
```

To reset sample-size to the default of 5 for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no sample-size
```

Related Commands

critical-interval
fail-count
normal-interval
ping-poll
show ping-poll
timeout (ping polling)
up-count

show counter ping-poll

This command displays the counters for ping polling.

Syntax `show counter ping-poll [<1-100>]`

Parameter	Description
<1-100>	A unique ping poll ID number. This displays the counters for the specified ping poll only. If you do not specify a ping poll, then this command displays counters for all ping polls.

Mode User Exec and Privileged Exec

Output **Figure 105-1: Example output from the show counter ping-poll command**

```
Ping-polling counters
Ping-poll: 1
PingsSent               ......... 15
PingsFailedUpState      ......... 0
PingsFailedDownState    ......... 0
ErrorSendingPing        ......... 2
CurrentUpCount          ......... 13
CurrentFailCount        ......... 0
UpStateEntered          ......... 0
DownStateEntered        ......... 0

Ping-poll: 2
PingsSent               ......... 15
PingsFailedUpState      ......... 0
PingsFailedDownState    ......... 0
ErrorSendingPing        ......... 2
CurrentUpCount          ......... 13
CurrentFailCount        ......... 0
UpStateEntered          ......... 0
DownStateEntered        ......... 0

Ping-poll: 5
PingsSent               ......... 13
PingsFailedUpState      ......... 0
PingsFailedDownState    ......... 2
ErrorSendingPing        ......... 2
CurrentUpCount          ......... 9
CurrentFailCount        ......... 0
UpStateEntered          ......... 0
DownStateEntered        ......... 0
```

Table 105-2: Parameters in output of the show counter ping-poll command

Parameter	Description
`Ping-poll`	The ID number of the polling instance.
`PingsSent`	The total number of pings generated by the polling instance.
`PingsFailedUpState`	The number of unanswered pings while the target device is in the Up state. This is a cumulative counter for multiple occurrences of the Up state.

Table 105-2: Parameters in output of the show counter ping-poll command(cont.)

Parameter	Description
PingsFailedDownState	Number of unanswered pings while the target device is in the Down state. This is a cumulative counter for multiple occurrences of the Down state.
ErrorSendingPing	The number of pings that were not successfully sent to the target device. This error can occur when your device does not have a route to the destination.
CurrentUpCount	The current number of sequential ping replies.
CurrentFailCount	The number of ping requests that have not received a ping reply in the current sample-size window.
UpStateEntered	Number of times the target device has entered the Up state.
DownStateEntered	Number of times the target device has entered the Down state.

Example To display counters for the polling instances, use the command:

```
awplus# show counter ping-poll
```

Related Commands
debug ping-poll
ping-poll
show ping-poll

show ping-poll

This command displays the settings and status of ping polls.

Syntax `show ping-poll [<1-100>|state {up|down}] [brief]`

Parameter	Description	
<1-100>	Displays settings and status for the specified polling instance.	
`state`	Displays polling instances based on whether the device they are polling is currently reachable or unreachable.	
	`up`	Displays polling instance where the device state is reachable.
	`down`	Displays polling instances where the device state is unreachable.
`brief`	Displays a summary of the state of ping polls, and the devices they are polling.	

Mode User Exec and Privileged Exec

Output **Figure 105-2: Example output from the show ping-poll brief command**

```
Ping Poll Configuration
-------------------------------------------------------------
Id Enabled State Destination
-------------------------------------------------------------
1  Yes     Down  192.168.0.1
2  Yes     Up    192.168.0.100
```

Table 105-3: Parameters in output of the show ping-poll brief command

Parameter	Meaning	
Id	The ID number of the polling instance, set when creating the polling instance with the **ping-poll** command.	
Enabled	Whether the polling instance is enabled or disabled.	
`State`	The current status of the device being polled:	
	`Up`	The device is reachable.
	`Down`	The device is unreachable.
	`Critical Up`	The device is reachable but recently the polling instance has not received some ping replies, so the polled device may be going down.
	`Critical Down`	The device is unreachable but the polling instance received a reply to the last ping packet, so the polled device may be coming back up.

Table 105-3: Parameters in output of the show ping-poll brief command(cont.)

Parameter	Meaning
Destination	The IP address of the polled device, set with the **ip (ping-polling)** command.

Figure 105-3: Example output from the show ping-poll command

```
Ping Poll Configuration
---------------------------------------------------------

Poll 1:
Description                          : Primary Gateway
Destination IP address               : 192.168.0.1
Status                               : Down
Enabled                              : Yes
Source IP address                    : 192.168.0.10
Critical interval                    : 1
Normal interval                      : 30
Fail count                           : 10
Up count                             : 5
Sample size                          : 50
Length                               : 32
Timeout                              : 1
Debugging                            : Enabled


Poll 2:
Description                          : Secondary Gateway
Destination IP address               : 192.168.0.100
Status                               : Up
Enabled                              : Yes
Source IP address                    : Default
Critical interval                    : 5
Normal interval                      : 60
Fail count                           : 20
Up count                             : 30
Sample size                          : 100
Length                               : 56
Timeout                              : 2
Debugging                            : Enabled
```

Table 105-4: Parameters in output of the show ping-poll command

Parameter	Description
Description	Optional description set for the polling instance with the **description (ping-polling)** command.
Destination IP address	The IP address of the polled device, set with the **ip (ping-polling)** command.

Table 105-4: Parameters in output of the show ping-poll command(cont.)

Parameter	Description	
`Status`	The current status of the device being polled:	
	`Up`	The device is reachable.
	`Down`	The device is unreachable.
	`Critical Up`	The device is reachable but recently the polling instance has not received some ping replies, so the polled device may be going down.
	`Critical Down`	The device is unreachable but the polling instance received a reply to the last ping packet, so the polled device may be coming back up.
`Enabled`	Whether the polling instance is enabled or disabled. The **active (ping-polling)** and **no active** commands enable and disable a polling instance.	
`Source IP address`	The source IP address sent in the ping packets. This is set using the **source-ip** command.	
`Critical interval`	The time period in seconds between pings when the polling instance has not received a reply to at least one ping, and when the device is unreachable. This is set with the **critical-interval** command.	
`Normal interval`	The time period between pings when the device is reachable. This is set with the **normal-interval** command.	
`Fail count`	The number of pings that must be unanswered, within the total number of pings specified by the **sample-size** command, for the polling instance to consider the device unreachable. This is set using the **fail-count** command.	
`Up count`	The number of consecutive pings that the polling instance must receive a reply to before classifying the device reachable again. This is set using the **up-count** command.	
`Sample size`	The total number of pings that the polling instance inspects when determining whether a device is unreachable. This is set using the **sample-size** command.	
`Length`	The number of data bytes to include in the data portion of the ping packet. This is set using the **length (ping-poll data)** command.	
`Timeout`	The time in seconds that the polling instance waits for a response to a ping packet. This is set using the **timeout (ping polling)** command.	
`Debugging`	Indicates whether ping polling debugging is **Enabled** or **Disabled**. This is set using the **debug ping-poll** command.	

Examples To display the ping poll settings and the status of all the polls, use the command:

```
awplus# show ping-poll
```

To display a summary of the ping poll settings, use the command:

```
awplus# show ping-poll brief
```

To display the settings for ping poll 6, use the command:

```
awplus# show ping-poll 6
```

To display a summary of the state of ping poll 6, use the command:

```
awplus# show ping-poll 6 brief
```

To display the settings of ping polls that have reachable devices, use the command:

```
awplus# show ping-poll state up
```

To display a summary of ping polls that have unreachable devices, use the command:

```
awplus# show ping-poll 6 state down brief
```

Related Commands **debug ping-poll**
ping-poll

source-ip

This command specifies the source IP address to use in ping packets.

By default, the polling instance uses the address of the interface through which it transmits the ping packets. It uses the device's local interface IP address when it is set. Otherwise, the IP address of the interface through which it transmits the ping packets is used.

The **no** variant of this command resets the source IP in the packets to the device's local interface IP address.

Syntax

```
source-ip {<ip-address>|<ipv6-address>}
no source-ip
```

Parameter	Description
<ip-address>	An IPv4 address in dotted decimal notation `A.B.C.D`
<ipv6-address>	An IPv6 address in hexadecimal notation `X:X::X:X`

Mode Ping-Polling Configuration

Examples To configure the ping-polling instance 43 to use the source IP address `192.168.0.1` in ping packets, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# source-ip 192.168.0.1
```

To configure the ping-polling instance 43 to use the source IPv6 address `2001:db8::` in ping packets, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# source-ip 2001:db8::
```

To reset the source IP address to the device's local interface IP address for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no source-ip
```

Related Commands

description (ping-polling)
ip (ping-polling)
length (ping-poll data)
ping-poll
show ping-poll

timeout (ping polling)

This command specifies the time in seconds that the polling instance waits for a response to a ping packet. You may find a higher time-out useful in networks where ping packets have a low priority.

The **no** variant of this command resets the set time out to the default of one second.

Syntax

```
timeout <1-30>
no timeout
```

Parameter	Description
<1-30>	Length of time, in seconds, that the polling instance waits for a response from the polled device.

Default The default is 1 second.

Mode Ping-Polling Configuration

Examples To specify the timeout as 5 seconds for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# timeout 5
```

To reset the timeout to its default of 1 second for ping-poll instance 43, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 43
awplus(config-ping-poll)# no timeout
```

Related Commands

critical-interval
fail-count
normal-interval
ping-poll
sample-size
show ping-poll
up-count

up-count

This command sets the number of consecutive pings that the polling instance must receive a reply to before classifying the device reachable again.

The **no** variant of this command resets the up count to the default of 30.

Syntax

```
up-count <1-100>
no up-count
```

Parameter	Description
<1-100>	Number of replied pings before an unreachable device is classified as reachable.

Default The default is 30.

Mode Ping-Polling Configuration

Examples To set the upcount to 5 consecutive pings for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# up-count 5
```

To reset the upcount to the default value of 30 consecutive pings for ping-polling instance 45, use the commands:

```
awplus# configure terminal
awplus(config)# ping-poll 45
awplus(config-ping-poll)# no up-count
```

Related Commands

critical-interval
fail-count
normal-interval
ping-poll
sample-size
show ping-poll
timeout (ping polling)

undebug ping-poll

This command applies the functionality of the **no debug ping-poll** command on page 105.6.

Chapter 106: sFlow Introduction and Configuration

sFlow Introduction 106.2
The sFlow Agent 106.3
    Sampling Methods 106.3
The sFlow Collector 106.5
Configuring sFlow on your Switch 106.5
    Configuration Procedure 106.6
    Configuration Example 106.7
sFlow Datagrams 106.12
The sFlow MIB 106.13

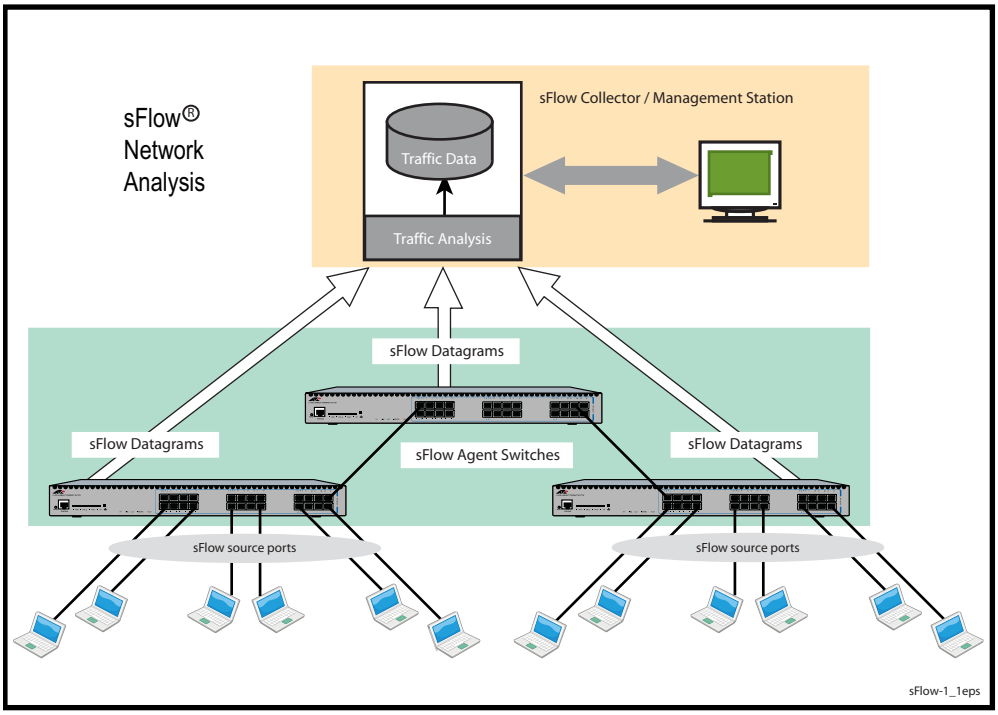
sFlow Introduction

sFlow®[1] provides the ability to monitor traffic in data networks containing switches and routers. A network employing sFlow typically comprises a number of network (sFlow) agents that accumulate sampled data and traffic counter information. The agents then forward this data to a collector. The collector then analyses the information supplied by its agents in order to compile and display statistical profiles of the network and its traffic. The sFlow feature on your switch provides the sFlow Agent capability.

1. sFlow® is a registered trademark belonging to InMon Corp, San Francisco, CA.

Figure 106-1 shows a basic sFlow network structure. The three network switches also function as sFlow agents. Each agent switch captures samples of the traffic passing through its monitored ports, and sends these samples together with counter information back to the sFlow collector. The agents sample data from a number of switch ports, each acting as an sFlow data source.

Figure 106-1: Basic sFlow network



The sFlow Agent

Your switch can act as an sFlow agent. The key capabilities of the agent are to:

- sample frames as they pass through selected ports on the switch, and provide sampled extracts of the network traffic.
- periodically capture interface counter data.
- package together the sampled frame and counter information that can be sent to the collector for analysis and display.
- be configurable via SNMP MIB objects.
- communicate to heterogeneous collector devices by means of standard protocols.

Agent components and functionality

sFlow functionality on your switch is based on the requirements defined in of RFC 3176 and its updates defined in the sFlow version 5 memo dated July 2004. This memo can be found at the web site, **www.sflow.org/sflow_version_5.txt.**

The terms defined in **Table 106-1** are used to describe the agent and its functionality on your switch:

Table 106-1: sFlow terminology

sFlow® Component	Definition
Network Device	Typically either a network switch or router that has the ability to forward frames across an Ethernet network; or between Ethernet networks, in the case of a router.
Data Source (sFlow Source Port)	The location of a sampling point within the switch. This is typically a switch port.
Packet Flow	The path taken by the data (frames) as they traverse a network device.
Sampling Rate	The ratio of frames passing through the data source, to those captured and forwarded as sFlow data. See **sflow sampling-rate** command on page 107.13.
Counter Sampling	The periodic polling of counters taken at the data source.
sFlow Datagram	A UDP datagram that contains details of sFlow captured data, and counters sent by the sFlow Agent to its Collector.

The sFlow agent (switch) uses sampling technology to derive traffic statistics from its monitored ports. Samples are taken at the sFlow source ports. After collecting its information, the switch then packetizes its samples and statistical data, and sends both to a remote sFlow collector.

Sampling Methods

Two sampling methods are employed within the sFlow agent, frame sampling, and counter sampling. Both sample types are combined within the datagrams sent to the collector. The frame sample data will result in a relatively constant traffic stream, but the counter information is sent where it can fill available space within each datagram. Datagrams are normally sent to the collector at the rate of one each second. However, several datagrams can be sent in rapid succession, where more information exists than can be sent in a single datagram.

Frame sampling

As frames enter or leave an sFlow source port, they are sampled at a rate determined by the **sflow sampling-rate** command on page 107.13 for that particular port.

Sampling occurs every N frames (on average), where N is the rate value set via **sflow sampling-rate** command. The sampling rate applies to ingress and egress frames independently. For example, a value of 1000, will sample one frame in every 1000 frames received, one in every 1000 frames sent from the specified port.

Caution

Setting the sFlow sampling rate to a very low value (frequent sampling) can place a heavy load on the switch's CPU. The severity of this loading will increase with the number of ports configured for sampling, the port speeds, and their data sampling rates.

Data confidentiality

Sampling operates by capturing the initial portion of frames (statistically) selected. The portion sampled is set by the **sflow max-header-size** command on page 107.10, or SNMP. If the **maximum header size** is greater than the actual headers in the sampled frames, then portions of the user data (payload) will also be captured and encapsulated in the datagrams sent to the collector. The amount of user data captured can be minimized by careful selection of the maximum header size.

Counter polling

The function of counter polling is to provide snapshots of various system counters. This produces a series of data counter sets for each port, which can be independently polled at user defined rates, and sent (once a second or less) to the collector. Allied Telesis switches running AlliedWare Plus software support generic interface counters only. For more information on the data types included in the sampling count, see **"sFlow Datagrams" on page 106.12**.

The sFlow Collector

The sFlow collector receives traffic samples and counter information from a number of sFlow agents. These samples are received as a series of UDP datagrams. From the data contained within these datagrams, the collector is able to provide statistical and or graphical information of network traffic.

The sFlow agent application on your switch supports only a single collector configuration.

sFlow collectors are proprietary third party products. Your switch, running as an sFlow agent has been designed for interoperability with any sFlow collector that supports the sFlow Version 5 specification, including the inMon sFlow collector.

The sFlow Collector may also contain an SNMP Manager that is able to configure sFlow on its agent switches.

Configuring sFlow on your Switch

This section provides some guidelines for setting up the sFlow® agent on your switch. sFlow can be configured directly on your switch - using the CLI, or it can be configured via an SNMP manager. The SNMP management function can be carried out either by a the sFlow collector, or a separate SNMP manager. The configuration examples in this section are shown using the CLI.

Caution

The sFlow configurations are set either by the switch's CLI, or the sFlow collector. Sometimes the collector will overide the sFlow settings that were initially configured by the CLI, in order to apply its own default settings.

If you want to apply the sFlow settings set by the CLI, or by an external network management system, then turn off network management at the collector.

We also advise that as part of your sFlow commissioning process, you review your security access procedures relating to sFlow access and its data traffic management.

sFlow configuration can vary greatly with your overall configuration, data profile, and monitoring intensity. Also, many interdependencies existing between parameter settings. For this reason, few firm configuration settings are recommended in this software reference, but instead these parameter relationships are explained and some typical configuration examples are shown.

The default settings on your switch have sFlow turned off for all ports.

The following commands are used to setup and configure sFlow on your switch. These are introduced in the order in which you would logically need to use them.

sFlow Command	Functionality
sflow enable	enables sFlow on your switch (or stack).
sflow max-header-size	sets the maximum sFlow data capture size.
sflow collector max-datagram-size	sets the maximum size for the agent to collector datagrams.
sflow agent (address)	sets the sFlow agent IP address on the switch.
sflow polling-interval	sets the counter polling interval for specified ports.
sflow sampling-rate	sets the mean sampling rate for specified ports.
sflow collector (address)	the sFlow agent's collector IP address and/or UDP port.

Configuration Procedure

The following process sets out a systematic procedure to configure sFlow on your switch:

Information gathering

sFlow configuration is dependent on your network structure and its data. Start by gathering together the following information.

- Obtain (or determine) the sFlow collector IP address.
- Select an appropriate UDP port for your sFlow datagrams. The recommended value is 6343, and is the default value preconfigured on your switch.
- Select an appropriate IP address for your sFlow agent. We recommend that you use the local IP address of your switch. For more information on local addresses and how to set them up, see the **interface (to configure)** command on page 14.3.
- Assess the sensitivity of the data that your sFlow agent will be sampling.
- Obtain details of the protocols that your sFlow agent will be sampling. If you intend sampling unusual or proprietary protocols, obtain details of their header lengths.
- Calculate the most appropriate max-header-size for your sFlow sampling.
- Select the ports that you want to sample, and their sample rate.

  These two factors vary (not quite) proportionally; so if you double the number of ports and double your sampling rate (i.e. sample half as many frames) then you will "almost" return to your earlier situation. Also note the speeds of the ports you have selected, because - for the same port utilization - the faster the port speed, the greater the load on the CPU.
- Review the speed of the port used to transport the sFlow datagrams to the collector. Unless configured to a specific port, the collector traffic will share the same network port with other traffic.

  The capacity of the collector port should be sufficient to carry the volume of sFlow traffic. This topic is expanded on in the **"Configuration Example" on page 106.7**.

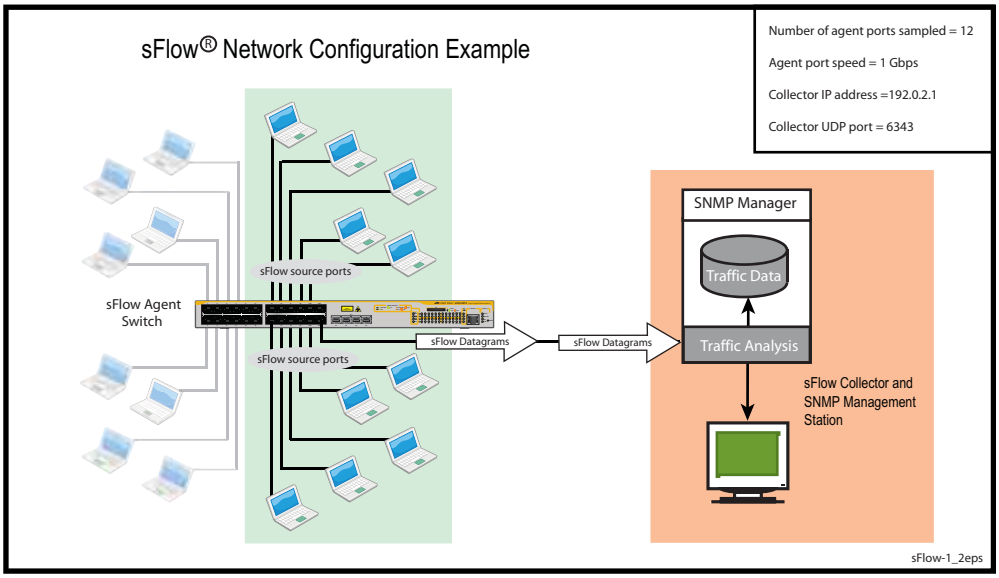
Managing the sFlow processing overhead

The sFlow data sampled on the ports converges into the CPU for processing and UDP packetizing. Therefore one of the major factors when configuring sFlow is to prevent the sFlow data volumes from placing a significant overhead on the CPU processing. The two most significant factors here are, the **number of ports sampled**, and the **sampling rate**. The other (and lessor) factors in this equation are the **frame size distribution** and the **maximum header size.** The shorter the frames are on the network, the heavier the sFlow processing load will be (for the same number of frames per second). Conversely the shorter the maximum header size selected, the lighter the sFlow processing load will be (because less data per frame is sent to the CPU).

Configuration Example

This section provides a configuration example based on the network shown below:

Figure 106-2: sFlow configuration example



Step 1: Determine the IP addresses and UDP ports.

Collector IP address is 192.0.2.65

sFlow UDP port uses the default of 6343

Agent (local) IP address 192.0.2.33. This is the address that the collector may use to configure the agent via SNMP.

Step 2: Determine the maximum sFlow datagram size.

Datagrams will be sent at one second intervals regardless of the amount of data they contain. If the amount of data to be sent is greater than the maximum datagram size, then several datagrams will be sent in quick succession - within the 1 second interval. The objective is to contain the sFlow information in a the minimum number of datagrams. That is. to fragment datagrams when necessary, but do it as little as possible.

Find the maximum datagram size that will pass through all network components without fragmenting. Then set the sFlow datagram size a little less than this value.

The maximum datagram size should be less than the MTU size.

For this example, the MTU is assumed to be set to its default of 1500 bytes. In this situation we could leave the maximum datagram size at its default of 1400 bytes; but in order to show this as a configuration step, we will change it to 1200 bytes.

Note

sFlow datagrams are generally transmitted at 1 second intervals. However, where there is more information than can fit into one datagram, several datagrams are sent sequentially, within the 1 second time frame.

Step 3: Determine the max-header-size for the sampled data.

The maximum header size for the sampled data is set by the **sflow max-header-size** command. The optimum setting is to capture only the header portion of the frame and discard the user-data portion. This is especially important where the user data contains sensitive information.

Keeping the max-header-size as small as possible has the additional benefit of lightening the CPU load.

First, inspect the nature of the data to be sampled and the protocols used to carry it.

For this example we will assume that the network contains Ethernet II frames with the 4 byte 802.1Q header component, IP, TCP protocols. In this situation the following rules can be applied:

For an environment using standard TCP/IPv4 over Ethernet frames, consider the following protocol basics.

Ethernet header (including the 4 byte 802.1Q header component) = 18 bytes

IPv4 header = 24 bytes

TCP header = 24 bytes

Total = 66 bytes

A similar calculation can be made for an environment using IPv6 over Ethernet.

Ethernet header (including the 4 byte 802.1Q component) = 18 bytes

IPv6 header = 40 bytes

TCP header = 24 bytes

Total = 82 bytes

Caution

In the above network scenarios:

- **For IPv4 - any data existing between 66 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-66=62 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.**
- **For IPv6 - any data existing between 82 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-82=46 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.**

For this example the **sflow max-header-size** will be set to 68 bytes (assuming an Ethernet - IPv4 - TCP environment).

Step 4: Select ports to sample.

Each sampled sFlow port speed is 1 Gbps

12 ports have been selected for sampling

Check that none of the ports selected have mirroring configured. sFlow will not operate if any port on the switch is configured for mirroring.

Step 5: Determine the sampling rate.

Selecting the sampling rate involves a trade-off between sFlow requirements, and system loading. The lower the sampling rate, the more samples will be taken, and the more accurate their results will be. Unfortunately, taking more samples

increases the load on the switch CPU and on the network connection to the collector.

For this particular configuration, the value of the sampling rate was set to 2750 to present a light load on the CPU.

Step 6: Review and adjust settings.

Because sFlow traffic loading will vary with the traffic profile, the following general assumptions are made. The following traffic profile is assumed.

- 50% of frames are <200 bytes long
- 40% of frames are >1400 bytes long

The following settings are:

- 12 x 1 Gbps ports are being sampled
- sFlow max-header size = 68 bytes
- sampling rate (N = 2750)
- average port utilization is assumed to be approximately 60%
- average data rate to the collector is assumed to be approximately 250 kbps

When setting the sampling rate, consider the following factors that will affect the CPU load. This load will increase (not necessarily linearly) as you:

- increase the number of ports configured
- increase the port speeds
- decrease the sampling rate
- increase the max-header-size

For this configuration the average sFlow collector traffic is expected to be approximately 250 kbps. In this example the agent-to-collector traffic will be shared with non-sFlow traffic. Although not described in this example, you can specifically configure the collector port to route only sFlow traffic. To do this you would need to assign a separate VLAN (and IP address) to the agent-to-collector interface and direct your sFlow traffic to this interface.

We advise that you ensure adequate bandwidth is provided for both the sFlow and general traffic that could share its network connection.

We will now use these settings to configure the network.

Configuration procedure

The following steps apply the settings obtained in the previous section.

Step 1: Enable sFlow switch-wide.

`awplus#`	
`configure terminal`	Enter Global Configuration Mode.
`awplus(config)#`	
`sflow enable`	Enable the sFlow agent globally on the switch.

Step 2: Configure the sFlow collector settings.

Step a. Set the sFlow collector max-datagram size	
`awplus(config)#`	
`sflow collector max-datagram-size 1200`	Set the maximum size of the sFlow datagrams to 1200 bytes.
Step b. Set the sFlow collector (address)	
`awplus(config)#`	
`sflow collector ip 192.0.2.65`	Set the sFlow collector address to 192.0.2.65.

Step 3: Configure the sFlow agent settings.

Step a. Set the sFlow agent (address)	
`awplus(config)#`	
`sflow agent ip 192.0.2.33`	Set the sFlow agent address to 192.0.2.33.
Step b. Set the sFlow sampling rate on sFlow Source Ports	
`awplus(config)#`	
`interface port1.0.11-port1.0.22`	Select the port range to configure (ports 1.0.11 to 1.0.22).
`awplus(config-if)#`	
`sflow sampling-rate 2750`	Set the sampling rate on the selected ports.

Step 4: Check the configuration.

`awplus(config-if)#`	
`do sh running-config sflow`	Validate that sFlow is enabled. **Note that the prefix "do" enables you to run an Exec Mode command from an Interface Mode prompt.**

Figure 106-3: Output from the show running-config sflow command

```
awplus#sh run sflow
!
sflow agent ip 192.0.2.33
sflow collector ip 192.0.2.65
sflow collector max-datagram-size 1200
sflow enable
!
interface port1.0.11-port1.0.22
 sflow sampling-rate 2750
```

sFlow Datagrams

After data sampling and counter information has been gathered, each sFlow agent packetizes the data and sends it to an sFlow collector where it can be analyzed and displayed in charts and tables.

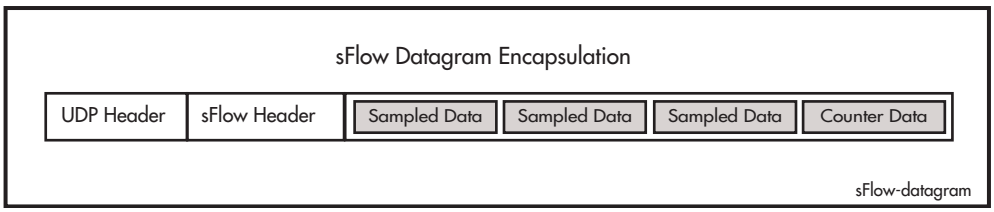
This packetized data is sent to the collector in UDP datagrams. These datagrams bear the IP address of the collector and the port number 6343. Using a standardized port helps to avoid configuration problems between the sFlow agents and collectors.

Although an analysis of the sFlow datagrams is outside the scope of this document, some basic information is provided here for those interested in knowing the basic components of the sFlow datagrams. The full specification of the sFlow protocol can be found at **www.sflow.org/sflow_version_5.txt**.

sFlow datagrams comprise three basic components:

- Datagram header information
- Flow sample information - may contain several samples
- Counter statistical information - fitted in where space permits

Figure 106-4: sFlow datagram encapsulation



The content of these datagram components is listed below:

sFlow header fields

- Version (The sFlow version being used)
- IP Address Type (Can be either an IPv4 or IPv6 address type)
- Source IP Address (The IP address of the sFlow agent)
- Sequence Number (The datagram sequence number)
- System Up-time
- Sample Count (The number of samples in the datagram)
- Sample Dataset

sFlow flow sample fields

- Flow Sample 1 (The first sample)
- Sample Type (Flow Sample, 0x0001)
- Sample Sequence Number (of flow samples)
- Sampler ID
- Sampling Rate (as set by the **"sflow sampling-rate" on page 107.13** or SNMP)
- Sample Pool (the total number of packets that could have been sampled)
- Packets Dropped (the number of packets dropped, due to a lack of resources)
- Input (the interface that the packet was received on - not supported)
- Output (the index number of the interface that the packet was sent from)

  (Note that your collector should have the ability via SNMP to resolve index numbers to physical port numbers)
- Packet Type
- Header Protocol - Ethernet ISO 88023(1)
- Packet Size (Frame Length including the FCS)
- Header Length - The sampled portion of the frame as set by the **"sflow max-header-size" on page 107.10**. May be shorter for small frames.
- Header Bytes

- Extended Elements Number
- Extended Elements

Note that in practice the Ethernet header is usually followed by components for the IP, TCP, and user data.

sFlow flow sample fields

- Counter Sample
- Sample Type (Counter Sample, 0x0002)
- 'Sample Sequence Number
- Sample ID (source ID index value)
- Sample Interval (as set by the "sflow polling-interval" on page 107.12)
- Counter Type (1=generic, 2=Ethernet)

Generic interface counters

- ifIndex
- ifType
- ifSpeed
- ifDirection (0=unknown, 1=full-duplex, 2=half-duplex, 3=in, 4=out
- ifStatus
- InOctets
- InUcastpackets
- InMulticast packets
- InBroadcast packets
- InDiscarded packets (= 0)
- InPackets containing errors
- InPackets containing unknown protocols (= 0)
- OutOctets
- OutUcast packets
- OutMulticast packets
- OutBroadcast packets
- OutDiscarded packets
- OutPackets containing errors
- ifPromiscuous Mode

Ethernet interface counters

- dot3Stats Alignment Errors (= 0)
- dot3Stats FCS Errors
- dot3Stats Single Collision Frames (= 0)
- dot3Stats Multiple Collision Frames
- dot3Stats SQE Test Errors
- dot3Stats Deferred Transmissions (= 0)
- dot3Stats Late Collisions
- dot3Stats Excessive Collisions
- dot3Stats Internal Mac Transmit Errors
- dot3Stats Carrier Sense Errors (= 0)
- dot3Stats Frame Too Longs
- dot3Stats Internal Mac Receive Errors
- dot3Stats Symbol Errors (= 0)

The sFlow MIB

Your switch fully supports inMon's sFlow MIB. For more information, see "sFlow-MIB" on page 95.80 and the website **www.sflow.org/SFLOW-MIB5.txt**.

Chapter 107: sFlow Commands

Command List 107.2
debug sflow 107.2
debug sflow agent 107.3
sflow agent (address) 107.4
sflow collector (address) 107.6
sflow collector max-datagram-size 107.8
sflow enable 107.9
sflow max-header-size 107.10
sflow polling-interval 107.12
sflow sampling-rate 107.13
show debugging sflow 107.14
show running-config sflow 107.15
show sflow 107.16
show sflow interface 107.17
undebug sflow 107.17

Command List

This chapter provides an alphabetical reference for sFlow commands.

debug sflow

This command enables sFlow® debug message logging, for sFlow sampling and polling activity on the specified ports. If no ports are specified, sampling and/or polling debug messages are enabled for all ports.

The **no** variant of this command disables sFlow sampling and or polling debug message logging on the ports selected. If no ports are specified, sampling and/or polling debug messages are disabled on all ports.

Syntax

```
debug sflow [interface <port-list>] [sampling][polling]
no debug sflow [interface <port-list>] [sampling][polling]
```

Parameter	Description
`interface`	Interface information.
`<port-list>`	The ports for which sFlow debug is to be enabled. The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.0.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.1.1-1.2.24`.
`sampling`	Debug sFlow sampling for the specified port(s).
`polling`	Debug sFlow polling for the specified port(s).

Default The sFlow sampling and or polling debug is disabled.

Mode Privileged Exec

Examples To enable sFlow debug message logging for polling and sampling on `port1.0.1` and `port1.0.7`, use the commands:

```
awplus# debug sflow interface port1.0.1,port1.0.7 sampling
       polling
```

To enable logging and polling of sFlow debug messages for polling and sampling on all ports, use the command:

```
awplus# debug sflow sampling polling
```

Related Commands **show debugging sflow**
no debug all

debug sflow agent

This command enables sFlow® debug message logging that is not specific to particular ports. For example, sending an sFlow datagram to the collector.

The **no** variant of this command applies the command default.

Syntax

```
debug sflow agent
```

```
no debug sflow agent
```

Default The sFlow agent debug message logging (that is not port specific) is disabled.

Mode Privileged Exec

Example To enable logging of sFlow agent debug messages, use the following command:

```
awplus# debug sflow agent
```

Related Commands

show debugging sflow
debug sflow

sflow agent (address)

This command sets the sFlow® agent IP address on the switch. This address is inserted into every sFlow datagram sent from the sFlow agent switch to the sFlow collector device. The sFlow collector can then uses this address to uniquely identify and to access the switch, such as for SNMP. We therefore recommend that you change this address as little as possible.

Although the agent address can be set to any valid IPv4 or IPv6 address; we recommended that you set the sFlow® agent IP address to be the **local address**[1] that is configured on the switch. This ensures that the sFlow collector can maintain connectivity to the switch irrespective of the addition or deletion of VLAN interfaces (each of which will have its own specific IP address). Note that sFlow is rendered inactive whenever the agent address is not set.

The **no** variant of this command applies its default setting to remove a configured address.

Syntax

```
sflow agent {ip <ip-address>|ipv6 <ipv6-address>}
no sflow agent {ip|ipv6}
```

Parameter	Description
<ip-address>	The IPv4 address of the switch that is acting as the sFlow agent.
<ipv6-address>	The IPv6 address of the switch that is acting as the sFlow agent. The IPv6 address uses the format X:X::X:X.

Default The sFlow agent address is unset.

Mode Global Configuration

Examples To set the sFlow agent (IPv4) address to `192.0.2.23`, use the command:

```
awplus# configure terminal
awplus(config)# sflow agent ip 192.0.2.23
```

To remove the sFlow agent (IPv4) address, use the command:

```
awplus# configure terminal
awplus(config)# no sflow agent ip
```

To set the sFlow agent (IPv6) address to `2001:0db8::1`, use the command:

```
awplus# configure terminal
awplus(config)# sflow agent ipv6 2001:0db8::1
```

1. **For information on local addresses and how to set them up, see the interface (to configure) command on page 14.3.**

To remove the sFlow agent (IPv6) address, use the command:

```
awplus# configure terminal
awplus(config)# no sflow agent ipv6
```

Related Commands **show running-config sflow**
show sflow

sflow collector (address)

This command sets the sFlow® agent's collector IP address and/or UDP port. This is the destination IP address and UDP port, for sFlow datagrams sent from the sFlow agent. The IP address can be any valid IPv4 or IPv6 address. Note that sFlow is rendered inactive whenever the collector address is set to 0.0.0.0 (for IPv4) or `::` (for IPv6).

The **no** variant of this command returns the IP address and UDP port values to their defaults, which will result in sFlow being deactivated.

Syntax

```
sflow collector {[ip <ip-address>|ipv6 <ipv6-address>]|
    [port <1-65535>]}

no sflow collector {[ip|ipv6]|[port]}
```

Parameter	Description
<ip-address>	IPv4 address of the remote sFlow collector.
<ipv6-address>	IPv6 address of remote sFlow collector. The IPv6 address uses the format X:X::X:X.
`port`	Destination UDP port for sFlow datagrams sent to the collector.
<1-65535>	UDP port number (default: 6343).

Default The collector address is `0.0.0.0` (which renders sFlow inactive), and the UDP port is `6343`.

Mode Global Configuration

Examples To set the sFlow collector address to `1920.2.25` and UDP port to `9000`, use the command:

```
awplus# configure terminal
awplus(config)# sflow collector ip 192.0.2.25 port 9000
```

To remove the sFlow collector IPv4 address and leave the UDP port unchanged, use the command:

```
awplus# configure terminal
awplus(config)# no sflow collector ip
```

To remove the sFlow collector IPv4 address and to remove the UDP port, use the command:

```
awplus# configure terminal
awplus(config)# no sflow collector ip port
```

To set the sFlow collector address to `2001:0db8::1` and leave the UDP port unchanged, use the command:

```
awplus# configure terminal
awplus(config)# sflow collector ipv6 2001:0db8::1
```

To remove the sFlow collector IPv6 address and leave the UDP port unchanged, use the command:

```
awplus# configure terminal
awplus(config)# no sflow collector ipv6
```

To remove the sFlow collector IPv6 address and to remove the UDP port, use the command:

```
awplus# configure terminal
awplus(config)# no sflow collector ipv6 port
```

Related Commands

show running-config sflow
show sflow

sflow collector max-datagram-size

This command sets the maximum size of the sFlow® datagrams sent to the collector.

The **no** variant of this command resets the maximum-datagram-size to the default.

Syntax

```
sflow collector max-datagram-size <200-1500>
no sflow collector max-datagram-size
```

Parameter	Description
<200-1500>	The maximum number of bytes that can be sent in an sFlow datagram sent from the agent to the collector.

Default 1400 bytes

Mode Global Configuration

Example To set the maximum datagram size to 1200, use the command:

```
awplus# configure terminal
awplus(config)# sflow collector max-datagram-size 1200
```

Related Commands

show running-config sflow
show sflow

sflow enable

This command enables sFlow® globally on the switch.

The **no** variant of this command disables sFlow globally on the switch.

Note that enabling sFlow does not automatically set its operational status to active. To activate sFlow the following conditions need to be met:

- sFlow is enabled.
- The sFlow agent address is set.
- The sFlow collector address is set to a valid (non zero) IPv4 or IPv6 address.
- Polling or sampling is enabled on the ports to be sampled or polled.

Syntax

```
sflow enable
no sflow enable
```

Default sFlow is disabled globally on the switch.

Mode Global Configuration

Example To enable sFlow operation, use the command:

```
awplus# configure terminal
awplus(config)# sflow enable
```

Related Commands **show running-config sflow**
show sflow

sflow max-header-size

This command sets the maximum header size of the Ethernet frames sampled on a specified port. The maximum header size is measured in bytes, referenced from the first byte of the Ethernet destination address and excludes the Ethernet FCS fields.

If a sampled Ethernet frame is longer than the maximum header size set by this command, then the frame will be truncated to the first N bytes before being placed in the sFlow datagram, where N is the maximum header size set by this command.

The **no** variant of this command resets the max-header-size to its default.

Syntax

```
sflow max-header-size <14-200>
no sflow max-header-size
```

Parameter	Description
<14-200>	The maximum number of header bytes to be sampled.

Default The max-header-size is 128 bytes.

Mode Interface Configuration

Usage The header size is measured from the first byte of the Ethernet frame MAC Destination Address.

For an environment using standard TCP IPv4 over Ethernet frames, consider the following basic protocol structure:

Ethernet header (including the 4 byte 802.1Q header component) = 18 bytes
IPv4 header = 24 bytes
TCP header = 24 bytes
Total = 66 bytes

See **"Determine the max-header-size for the sampled data." on page 106.8** for more information on configuring this command.

A similar consideration can be made for an environment using TCP IPv6 over Ethernet:

Ethernet header (including the 4 byte 802.1Q header component) = 18 bytes
IPv6 header = 40 bytes
TCP header = 24 bytes
Total = 82 bytes

Caution

In the above network scenarios:

- **For IPv4 - any data existing between 66 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-66=62 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.**
- **For IPv6 - any data existing between 82 bytes and the value set by this command will be included in the sFlow packet samples. For example, with the default of 128 applied, up to 128-82=46 bytes of user data could be included in the sFlow datagram samples sent between the Agent and the Collector.**

Note that the agent-to-collector datagrams contain their own UDP headers, which are outside this calculation.

Example To set the maximum header size to 160 bytes for ports 1.0.1 and 1.0.7, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# sflow max-header-size 160
```

Related Commands

show running-config sflow
show sflow interface
sflow max-header-size

sflow polling-interval

This command sets the sFlow® counter polling interval (in seconds) for the specified ports. A value of 0 disables polling. A counter sample is taken every N seconds where N is the value set by this command.

The **no** variant of this command applies the default.

Syntax

```
sflow polling-interval {0|<1-16777215>}
no sflow polling-interval
```

Parameter	Description
0	Disable polling (the default).
<1-16777215>	The polling interval in seconds.

Default The polling-interval is 0 (polling disabled).

Mode Interface Configuration

Example To set the polling interval to 60 seconds for ports 1.0.1 and 1.0.7, use the following commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# sflow polling-interval 60
```

Related Commands

show running-config sflow
show sflow interface

sflow sampling-rate

This command sets the mean sFlow® sampling rate for the specified ports. Sampling occurs every N frames (on average), where N is the rate value set via this command. The sampling rate applies to ingress and egress frames independently. For example, a value of 1000 will sample one frame in every 1000 frames received, i.e. one in every 1000 frames sent from the specified port. A value of 0 disables sampling on the specified port(s).

The **no** variant of this command applies the default.

Syntax

```
sflow sampling-rate {0|<256-16777215>}
no sflow sampling-rate
```

Parameter	Description
0	Sets the default.
<256-16777215>	The sampling rate N, measured in Ethernet frames.

Default The sampling-rate is 0 (sampling disabled).

Mode Interface Configuration

Example To set the sampling rate to 500 for ports 1.0.1 and 1.0.7, use the commands:

```
awplus# configure terminal
awplus(config)# interface port1.0.1,port1.0.7
awplus(config-if)# sflow sampling-rate 500
```

Related Commands

show running-config sflow
show sflow interface

show debugging sflow

This command displays sFlow® debug settings for agent operation, and for sampling and polling on specific interface ports. If no interface ports are specified, sampling and polling will be applied to all ports.

Syntax

```
show debugging sflow [interface <port-list>]
```

Parameter	Description
`interface`	The interface information.
`<port-list>`	The ports for which the sFlow debug settings are to be shown. The ports to display information about. The port list can be: ■ a switch port (e.g. `port1.1.12`) ■ a continuous range of ports separated by a hyphen, e.g. `port1.0.1-1.0.24` ■ a comma-separated list of ports and port ranges, e.g. `port1.0.1,port1.1.1-1.2.24`.

Mode User Exec and Privileged Exec

Example To display sFlow debug settings on the agent, and for sampling and polling on ports 1.0.1 to 1.0.9, use the command:

```
awplus# show debugging sflow interface port1.0.1-1.0.9
```

Output **Figure 107-1: Sample obtained for an sFlow agent**

```
awplus# show debugging sflow interface port1.0.1-1.0.9

sFlow Agent Debug:    Enabled

           Sampling       Polling
Port        Debug          Debug
------------------------------
1.0.1      Enabled        Enabled
1.0.2      Enabled           -
1.0.3         -              -
1.0.4         -              -
1.0.5         -              -
1.0.6         -           Enabled
1.0.7         -              -
1.0.8         -           Enabled
1.0.9         -           Enabled
```

To display sFlow debug settings for all ports, use the command:

```
awplus# show debugging sflow
```

Related Commands
show running-config sflow
show sflow interface

show running-config sflow

This command displays the running system information specific to the sFlow feature.

Syntax `show running-config sflow`

Mode Privileged Exec and Global Configuration

Example To display the sFlow running configuration information, use the command:

```
awplus# show running-config sflow
```

Output **Figure 107-2: Example output from the show running-config sflow command**

```
awplus#sh run sflow
!
sflow agent ip 192.0.2.33
sflow collector ip 192.0.2.65
sflow collector max-datagram-size 1200
sflow enable
!
interface port1.0.11-port1.0.22
 sflow sampling-rate 512
```

Related Commands show running-config

show sflow

This command displays non-port-specific sFlow agent configuration and operational status.

Syntax `show sflow`

Mode Privileged Exec

Example To display sFlow configuration and operational status, use the command:

```
awplus#  show sflow
```

Output

Figure 107-3: Example output from the show sflow command

```
sFlow Agent Configuration:                    Default Values
  sFlow Admin Status ........ Disabled        [Disabled]
  sFlow Agent Address ....... [not set]       [not set]
  Collector Address ......... 0.0.0.0         [0.0.0.0]
  Collector UDP Port ........ 6343            [6343]
  Tx Max Datagram Size ...... 1200            [1400]

sFlow Agent Status:
  Polling/sampling/Tx ....... Inactive because:
                                - sFlow is disabled
                                - Agent Addr is not set
                                - Collector Addr is 0.0.0.0
                                - Polling & sampling disabled
                                  on all ports
```

Table 107-1: Parameters in the output of the show sflow command

Output Parameter	Description
`sFlow Admin Status`	Whether sFlow agent operation is administratively enabled.
`sFlow Agent Address`	The sFlow agent IPv4 or IPv6 address for the device. sFlow is rendered inactive whenever the agent address is not set.
`Collector Address`	The IPv4 or IPv6 collector address to which sFlow datagrams are sent. sFlow is rendered inactive whenever the collector address is set to 0.0.0.0 or 0:0::0.0.
`Collector UDP Port`	The UDP port on the collector to which sFlow datagrams are sent.
`Tx Max Datagram Size`	The maximum size of the sFlow datagrams sent to the collector.
`Polling/sampling/Tx`	Whether sFlow sampling and/or polling (and hence sFlow datagram transmission) are active. If inactive the reasons are listed.

Related Commands show running-config sflow
show sflow interface

show sflow interface

This command displays sFlow agent sampling and polling configuration for specified ports.

Syntax `show sflow interface <ifrange>`

Parameter	Description
<ifrange>	The interface range.

Mode Privileged Exec

undebug sflow

This command applies the functionality of the **no debug sflow** command.

Chapter 108: Cable Fault Locator Introduction

Introduction to the Cable Fault Locator 108.2
Capabilities 108.2
TDR Operating Principles 108.2
Using the Cable Fault Locator 108.3

Introduction to the Cable Fault Locator

The Cable Fault Locator (CFL) is a cable diagnostic tool located within the switch. For a selected port, the CFL will display connection status or faults that exist in either the connecting cable itself, or its terminations.

Capabilities

The CFL is designed to operate on cable systems that utilize the following:

- fixed copper ports, i.e. not using SFP type pluggable transceivers.
- unshielded twisted pair data cables such as CAT 5 or CAT 6 and up to 100 meters long.
- cable terminations that use RJ-45 or RJ-0.5 connections.
- data rates from 10 Mbps to 1 Gbps (10 Gbps over copper cable is not supported).

The CFL operates using a technology known as Time Domain Reflectometry (TDR) to test all four pairs of wires inside the cable.

Note that CFL cannot run on a port that is currently supplying power via PoE.

TDR Operating Principles

When a data cable is correctly terminated, the data energy traveling along it is absorbed by its terminating load resistance. However, if the cable is unplugged, broken or short circuited, this energy is reflected at the cable termination and travels back along the cable towards its source connection.

To test the cable, the CFL generates a pulse at the cable source connection and monitors the cable for the presence of a returning (reflected) pulse. By measuring the timing between the transmitted and reflected pulses, the CFL can calculate the distance between a fault (usually at the distant termination) and cable's source connection. Also, by detecting the polarity of the reflected signal, the CFL can determine whether the fault is due to an open circuit, or a short circuit, condition.

Using the Cable Fault Locator

To run a CFL diagnostics test, use the command, **"test cable-diagnostics tdr interface" on page 109.3**. This will return a prompt asking you to confirm whether or not you want to continue with the test. This confirmation prompt is generated because the link is unable to carry data during the test, typically taking between 1 and 2 seconds to complete.

Example To run a CFL test on the cable inserted into port 1.0.1 use the following command:

```
awplus# test cable-diagnostics tdr interface
        port1.0.1
```

This command returns the following message:

```
Link will go down while test is in progress. Continue? (y/n):
```

Select y to continue.

```
awplus# y
```

Answering y returns the following message:

```
Test started. This will take several seconds to complete. Use
"show test cable-diagnostics tdr" to print results.
```

Once the cable test has run you can display its results by running the **show test cable-diagnostics tdr** command on page 109.2.

Output **Figure 108-1: Example output from the show test cable-diagnostics tdr command**

```
Port     Pair Length       Status
-----------------------------------------
1.0.1    A    -            OK
         B    -            OK
         C    5 +/- 5 m    Open
```

From the monitoring tests described, the CFL presents cable termination status information as shown, together with explanations, in **Table 108-1**.

Table 108-1: Cable status table

Status	Definition
OK	The pair is good and is terminated.
Open	The pair is not terminated.
Short (within-pair)	There is a short between the two wires of the pair.
Short (cross-pair)	There is a short between wires of different pairs.
Error	The test was unable to get a result. This error condition may occur when connecting to remote devices that issue idle traffic data when operating in the 10/100 M mode.

Chapter 109: Cable Fault Locator Commands

Command List 109.2
clear test cable-diagnostics tdr 109.2
show test cable-diagnostics tdr 109.2
test cable-diagnostics tdr interface 109.3

Command List

This chapter provides an alphabetical reference of commands used to test copper cables. For more information, see Cable Tester Introduction.

clear test cable-diagnostics tdr

This command clears the results of the last cable test that was run.

Syntax `clear test cable-diagnostics tdr`

Mode Privileged Exec

Examples To clear the results of a previous cable-diagnostics test use the following commands:

```
awplus# clear test cable-diagnostics tdr
```

show test cable-diagnostics tdr

This command displays the results of the last cable-diagnostics test that was run using the TDR (Time Domain Reflectometry) on a fixed copper cable port.

The displayed status of the cable can be either:

- OK
- Open
- Short (within-pair)
- Short (across-pair)
- Error

Syntax `show test cable-diagnostics tdr`

`no enable`

Mode Privileged Exec

Examples To show the results of a cable-diagnostics test use the following command:

```
awplus# show test cable-diagnostics tdr
```

Output **Figure 109-1: Example output from the show test cable-diagnostics tdr command**

```
Port      Pair Length       Status
------------------------------------------
1.1.1     A    -            OK
          B    -            OK
          C    5 +/- 5 m    Open
```

test cable-diagnostics tdr interface

This command applies the Cable Fault Locator's cable-diagnostics tests to twisted pair data cables for a selected port. The tests will detect either correct, short circuit, or open, circuit terminations. For more information on running the CFL, see the **Cable Fault Locator Introduction** chapter.

The test can take several seconds to complete. See the related show command to display the test results.

A new test can only be started if no other test is in progress. CFL cannot run on a port that is currently supplying power via PoE.

The displayed status of the cable can be either, OK, Short (within-pair), or Open. The "Open" or "Short" status is accompanied with the distance from the source port to the incorrect termination.

Syntax `test cable-diagnostics tdr interface <interface>`

Parameter	Description
`cable-diagnostics`	The cable diagnostic tests.
`tdr`	Time Domain Reflectometry.
`interface`	Selects the interface to test.
`<interface>`	Interface number of the port to be tested, i.e. 1.0.2.

Example To run a cable test on the cable inserted into port 1.0.1 use the following command:

```
awplus# test cable-diagnostics tdr interface
port1.0.1
```

You will receive the following message:

```
Link will go down while test is in progress. Continue? (y/n):
y Select y to continue.
```

```
awplus# y
```

You will then receive the following message:

```
Test started. This will take several seconds to complete. Use
"show test cable-diagnostics tdr" to print results.
```

Part 8: Virtual Chassis Stacking

- Chapter 110 VCStack Introduction
- Chapter 111 Stacking Commands

Chapter 110: VCStack Introduction

VCStack Introduction 110.2
- Features of Virtual Chassis Stacking 110.2
- VCStack Capable Switches 110.2
- The Physical Stack 110.3
- Two-Switch Stack Configuration 110.3
- Multiple Switch Stack Configuration 110.4
- Resilient Stacked Topology 110.4

Stack Formation 110.8
- The Role of the Stack Master 110.8
- Stack Management VLAN 110.9

Stack Member Failure and Recovery 110.12
- Fixed or Virtual MAC Addressing 110.12
- Stack Resiliency Link 110.13
- Stack Failure Recovery 110.14
- Stack Separation and Recovery 110.15
- Stack Maintenance 110.15

Disabled Master Monitoring (DMM) 110.17

Provisioning (Stack Members) 110.19
- Provisioned Board Classes 110.19
- Applying Hardware Provisioning 110.19
- Removing Hardware Provisioning 110.21
- Displaying Provisioned Configurations 110.22
- Provisioning and Configuration Management 110.23

Software Version Auto Synchronization 110.24
- Introduction 110.24
- How Auto Synchronization Works 110.24

Stack License Management 110.27

VCStack Introduction

This chapter describes Virtual Chassis Stacking (VCStack), its features, and basic connection examples. For detailed descriptions of the commands used to configure VCStack, see **Chapter 111, Stacking Commands**. Also, see **Allied Telesis Support Documentation** for detailed instructions on how to install this feature.

VCStack is a group of physically separate switches that are configured to operate as a single logical switch. In order to function as a VCStack, its component switches are connected using high-speed stacking links.

Features of Virtual Chassis Stacking

Creating a VCStack greatly eases network management, because you can configure all the stacked devices via a single IP address. Creating a VCStack will often eliminate your need to configure protocols such as VRRP and Spanning Tree. VCStack also enables you to create highly resilient networks. This resiliency can be applied in several ways.

Within the stack itself, switch interconnection is via two links. The second link is able to provide an alternative data path, thus the stack will continue to function if a single switch fails. Degraded performance might occur however, due to the reduced VCStack bandwidth.

User ports can also be made extremely resilient by utilizing link aggregation. Aggregated links can span ports, modules, and even switches within the stack. Creating aggregated links that span multiple switches within a stack creates an extremely resilient configuration. Communication will still exist even if a switch and its aggregated ports fail. Refer to **Figure 110-4 on page 110.5.**

Caution

Stack operation is only supported if stack virtual-mac is enabled.
For more information refer to:
"Enabling the stack virtual-mac" on page 110.12 and
"stack virtual-mac" on page 111.32

VCStack Capable Switches

VCStack is supported on the following Allied Telesis switch types:

- SwitchBlade® x8100 Series (VCStack Plus)
- x900-24XT, x900-24XS, x900-24XT-N
- x900-12XT/s
- SwitchBlade® x908
- DC2552XS
- x610 Series
- x510 Series
- IX5-28GPX
- x310 Series

Note

You can only create VCStacks using switches from within the same product group, for example, or all x610 Series switches.

Stacking connectivity and functionality varies slightly between switch types.Your x510 Series switch can support a maximum of four devices per stack. Consult the appropriate software reference for stacking functionality of other Allied Telesis switches.

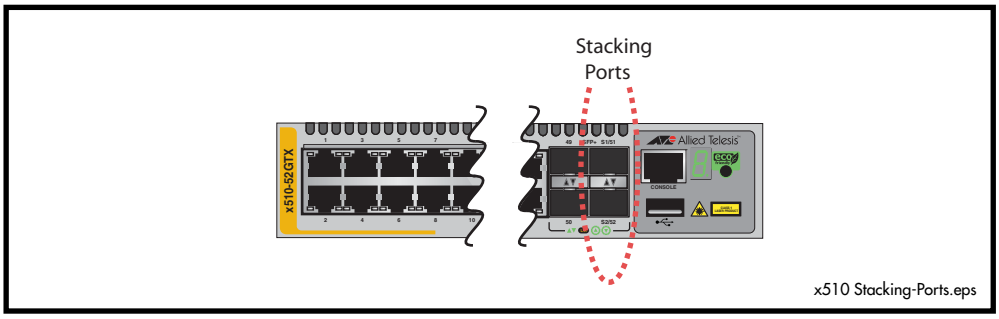
The Physical Stack

A VCStack can consist of up to four individual stack members interconnected via high-speed stacking links (SFP+ ports 27 and 28 on 28 port switches, and ports 51 and 52 on 52 port switches). As the stack forms, its switch members elect one of them to become the primary stack member called the stack master (displayed in the show commands as the active master). The remaining switches then become ordinary members of the stack, and are referred to as backup members.

VCStack cables and connections

Stack members are interconnected via the SFP+ ports shown in **Figure 110-1**. Note that stacking cables should connect from port SFP/51 on one stack member, to port SFP/52 on the other (or from port SFP/27 on one stack member, to port SFP/28 on the other)

Figure 110-1: Stacking ports

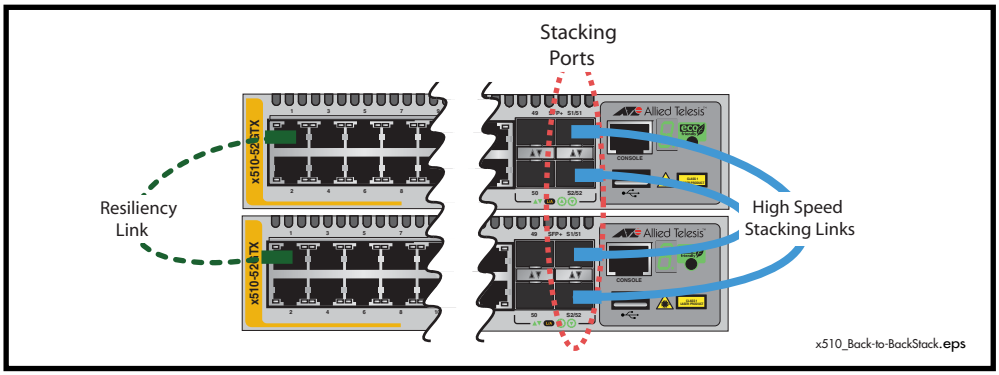


Two-Switch Stack Configuration

Ring configuration using two switches

This configuration, shown in **Figure 110-2**, uses two switches that are connected back to back via two high-speed stacking links. Note that stacking ports labeled S1/27 must connect to stacking ports labeled S2/28. Although in this configuration the stack can still function using only a single high-speed stacking link, we recommend using both stacking links as shown.

Figure 110-2: Back-to-back topology

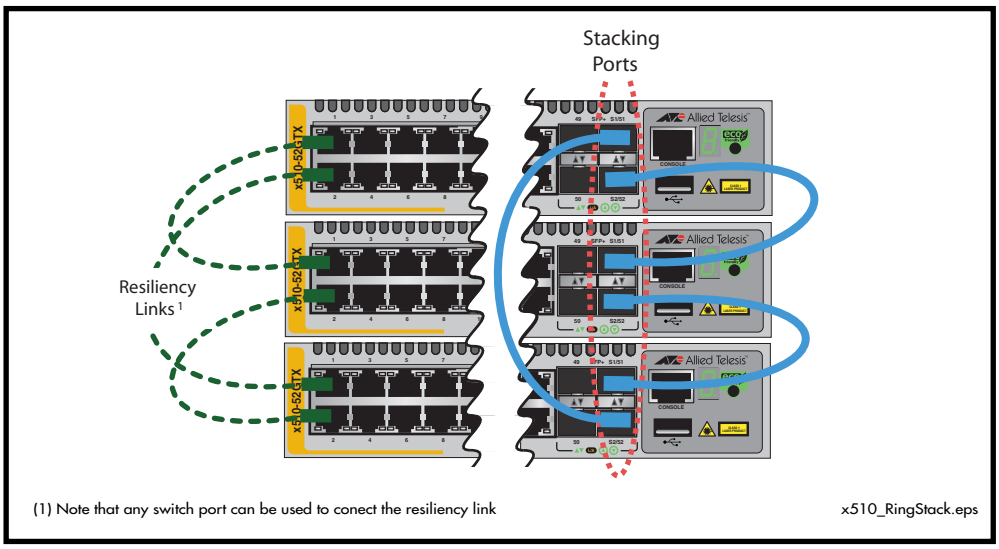


Multiple Switch Stack Configuration

Ring configuration using multiple switches

A VCStack using x510 switches can comprise up to four stack members connected in a ring topology. **Figure 110-3** shows a ring comprising three stacked x510 series switches. Because alternate paths are provided between the stack members' stacking links, this topology offers a very resilient configuration.

Figure 110-3: VCStack ring topology using x510 switches

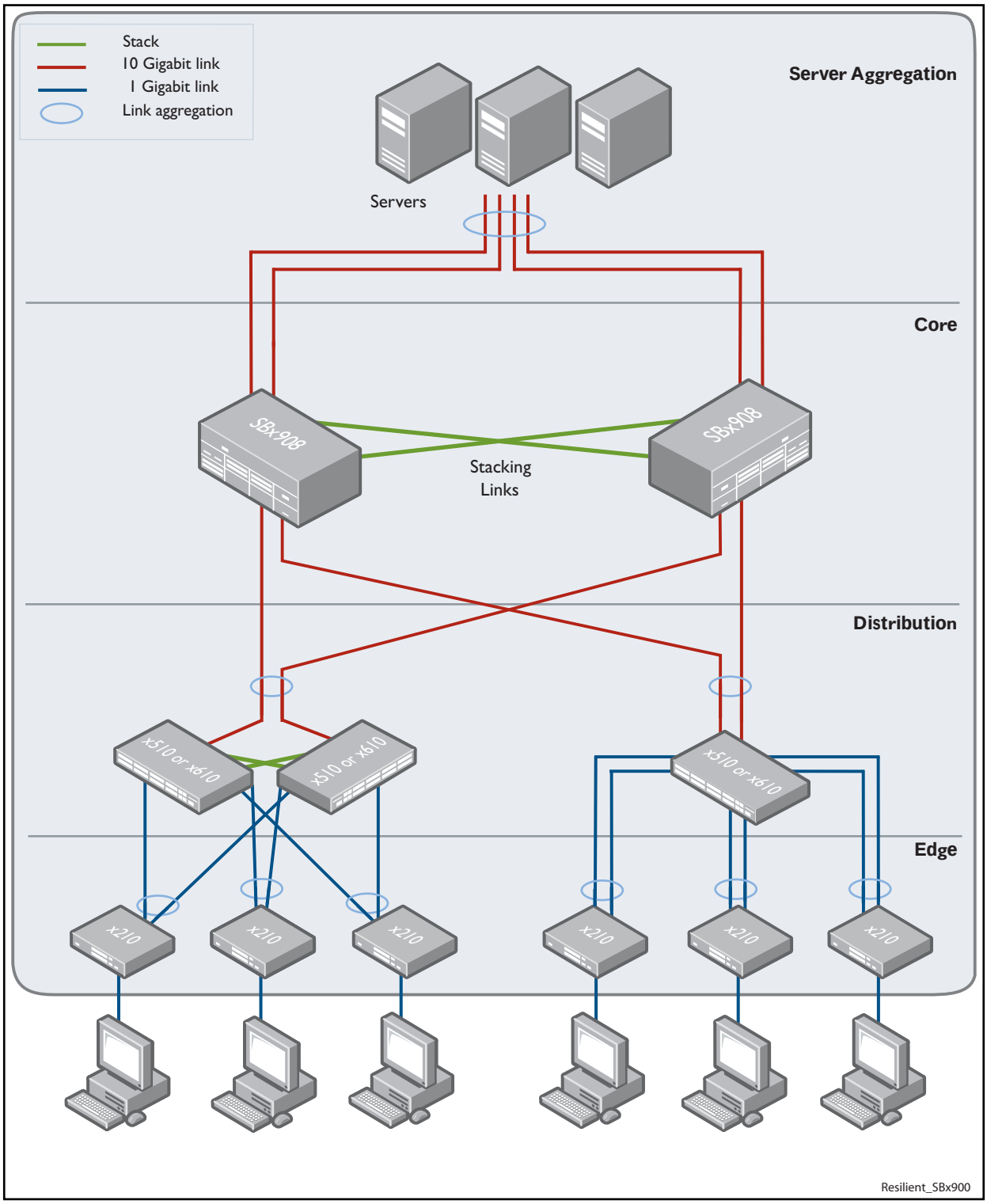


Resilient Stacked Topology

Where network connectivity uptime is a major criteria, you can use virtual chassis stacking to create highly reliable network configurations. The network shown in **Figure 110-4** employs redundant links and switches to create a stacked network that offers extremely reliable user connectivity.

Employing link aggregation rather than spanning tree to manage the parallel paths enables the bandwidth of both data links to be utilized under normal conditions, whilst enabling a single data link to operate should its partner link fail.

Figure 110-4: VCStack resilient stacked topology example



The network depicted in the above figure employs two SwitchBlade® x908 switches to form an expandable network core. These switches are stacked and so appear as a single logical switch (note that smaller switches such as can be also be used to form the stacked core or stacked distribution).

This network topology supplies multiple dual connections to a number of downstream distribution switches that can in turn connect to user devices. Similarly, the dual network paths provide very reliable connectivity to the servers portion of the network.

Resiliency link

The resiliency link carries no network data. Its function is to provide additional stack status information to enable the stack members to more accurately decide whether it is appropriate for one of them to take over the role of stack master if the existing master fails. See **"Stack Resiliency Link" on page 110.13**.

Resiliency link configurations via switch ports

Two resiliency-link configurations that use switch ports are shown below. The first figure shows the resiliency link connecting in a ring topology, whilst the second figure shows the resiliency link connecting to the switch ports via a network hub. In both configurations, the resiliency link connections are made using a designated VLAN running over switch-port connections between each stack member. For more information on using the resiliency link commands see the **stack resiliencylink** command on page 111.28 and the **switchport resiliencylink** command on page 111.34.

Figure 110-5: Resiliency link connecting to switch ports over the ResiliencyLink VLAN

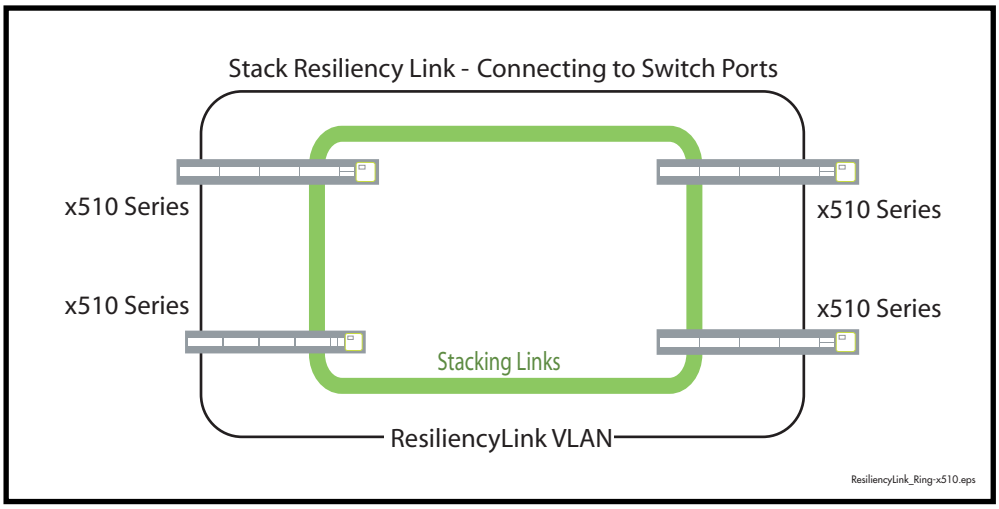
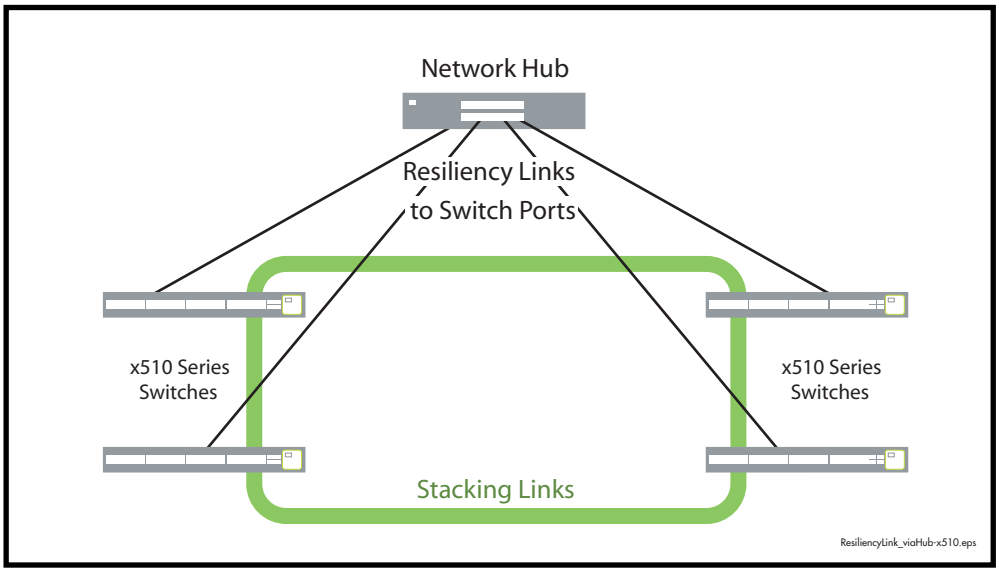


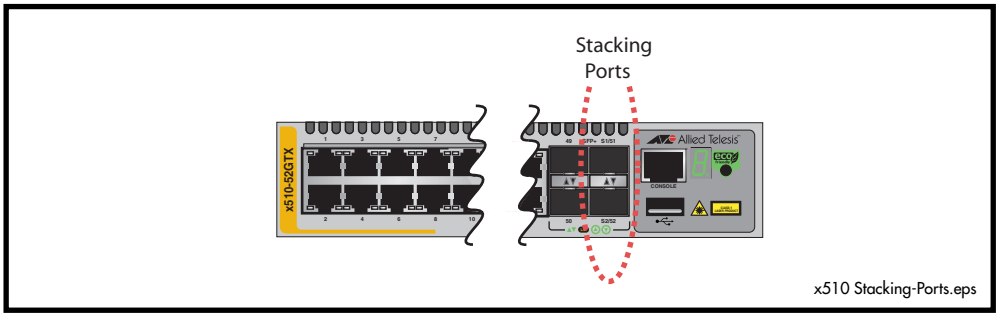
Figure 110-6: Resiliency link connecting to switch ports over the ResiliencyLink VLAN using a network hub



Stack Formation

As previously mentioned, a VCStack always contains a stack (active) master plus a number of stack (backup) members. To be part of a stack, a switch must connect to the other potential stack members via dedicated stacking ports. See **Figure 110-7**.

Figure 110-7: Stacking ports



Once the switches have been physically connected to form a stack, powering all the members on automatically sets off a number of processes that enable the stack members to detect the presence of the other stack members and form themselves into a VCStack.

Long distance stacking

You can extend the distance between stacked units up to the maximum supported by the particular SFP+ you are using. This capability enables you to create a stack of up to 4 geographically separated switches as a single stack.

The Role of the Stack Master

In addition to being a member of its VCStack, the stack master manages functions such as software version control and distribution, routing processing, and network management.

Selecting the stack master

The stack members are able to automatically select which switch will become the stack master. This selection is based on two components:

1. The stack member's priority setting.
2. The stack member's MAC address.

For both components, the lower the number the higher the priority will be. To set the stack priority, run the **stack priority** command on page 111.25. Note that changes to these settings will not take effect until the next master re-election. To display these components run the **show stack** command on page 111.15.

The master is the switch with the lowest priority setting, or if the priority settings are equal, the switch with the lowest MAC address will become the stack master. When a stack member is initially booted, its priority value defaults to 128. Therefore, if all switches retain their defaults, then the stack master will be determined by MAC address comparison.

The stack also assigns a stack-ID number to each member. This number provides a unique reference number for switches within the stack; it plays no part in selecting the stack master. The stack-ID is used as the first digit of the three component port identifier numbers. For example, port number 2.0.14 has the stack-ID of 2.

Note

This last point is an important one to remember when using configuration scripts. You should ensure that you modify your configuration scripts to match any changes you have made to the stack-ID assignments.

Note

The ability to independently set both a stack member's priority and its ID means that the stack master does not need to have an ID of 1, although configuration is simplified by arranging for ID 1 to be the device with the lowest priority value - and thereby forcing it to be the stack master. If you create a stack using new switches, the following (simplified) process should ensure that the master member has an ID of 1.

New switches are shipped with a Stack Member-ID of 1 and a priority of 128. If four such switches are created as a stack, the switch with the lowest MAC address will be selected to be the stack master (because all priority settings are 128). The remaining three stack member devices will then reboot. The stack master does not reboot and retains its Stack Member-ID of 1.

You can change the stack-ID by using the **stack renumber** command on page 111.26.

Common stack configuration

Once the switches have configured themselves into a VCStack, they all share the same configuration information and startup scripts.

Stack Management VLAN

Managing the stack is the same as managing an individual switch. You can connect to the asynchronous console port of any stack member, or you can set an IP address on a network VLAN (for example, VLAN 1) and use SSH for remote access.

As the switches form themselves into a stack, each switch creates a common stack management VLAN and a management IP address. Both the VLAN ID and the IP address are internal entities that are used between the stacked switches, via the stacking ports, and therefore do not appear on the user network.

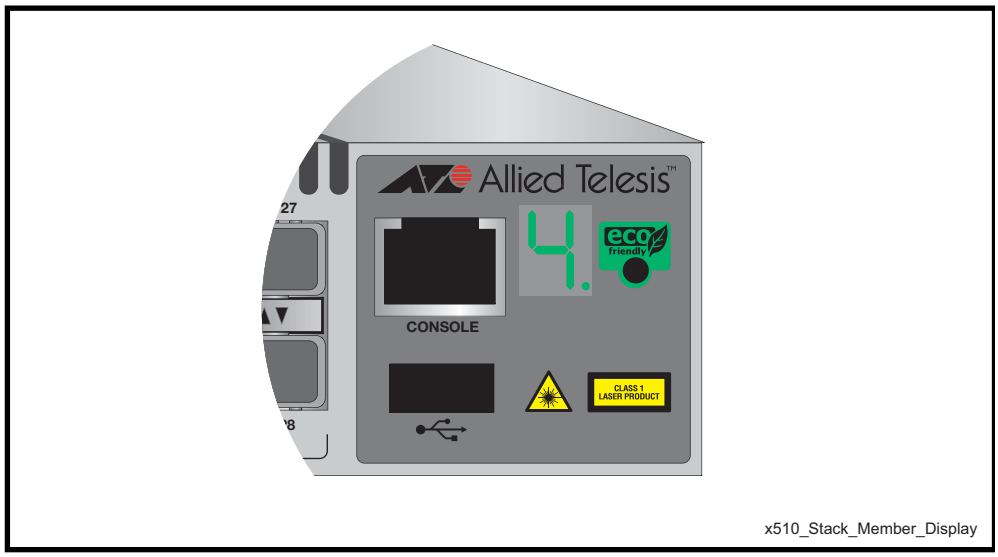
Initially the stack assigns the default VLAN tag ID of 4094 to the management VLAN, and assigns an IP address from the subnet 192.168.255.0/27 to this VLAN as the management IP address. Once the stack has formed, you can change both these settings. To change the VLAN ID use the **stack management vlan command on page 111.24**. To change the management IP address use the **stack management subnet command on page 111.23**. Note however, that you must keep the 27 bit subnet mask, (/27 or 255.255.255.224).

Also note that the management VLAN ID and management IP subnet must be unique across the stack's internal and external network. This means you cannot use the management VLAN ID or management IP subnet elsewhere in the user network. However, note that stacks in the same network can all use the same management VLAN ID and management IP subnet settings internally. To view the current settings for the stack management VLAN ID and IP address, use the **show stack command on page 111.15**.

Stack member identification

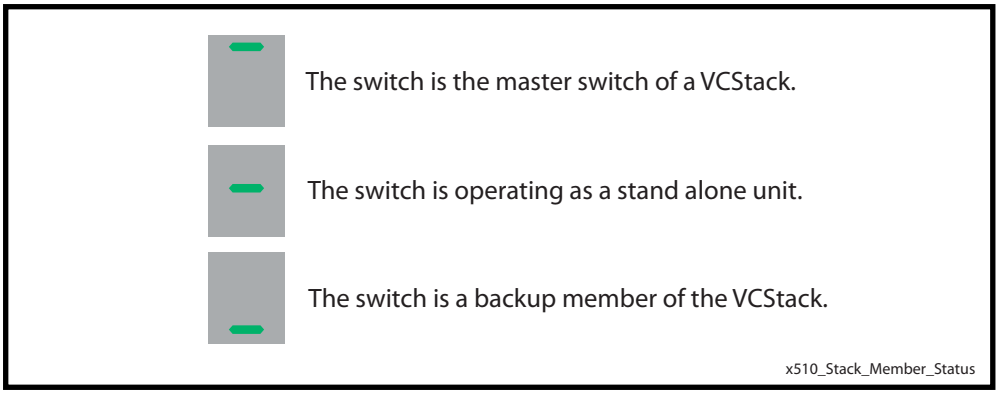
When a switch becomes a stack member, it is assigned a Stack Member ID. A numeric ID LED on the front panel indicates its stack member ID number. See **Figure 110-8 on page 110.10**. Switches in a VCStack usually have stack IDs in the range 1 to 4 (although any four numbers between 1 and 8 can be assigned). A stand-alone switch displays the ID number 0.

Figure 110-8: Stack member ID display



In addition, depressing the “eco” button on this panel will change the LED display to indicate the status of the switch within a stack, or whether it is operating as a stand alone switch. See **Figure 110-8 on page 110.10**.

Figure 110-9: Stack_Member_Status display



Running commands on specific stack members

In some situations, you may want to obtain information that is specific to a particular stack member. For many **show** commands, you can specify the desired stack member. For example, to show the processes for stack member 2, use the following command:

```
awplus# show process 2
```

In other cases, you can use the **remote-login** command to log into the specific stack member. For example, to see a directory listing for stack member 2, use the following commands:

```
awplus# remote-login 2
awplus-2# enable
awplus-2# dir
```

To return to the command prompt on the master stack member, type **exit**.

For more information on using this command, see the **remote-login command on page 111.7**.

Running QoS within a VCStack

When you configure QoS on a VCStack, you need to make the following changes:

Switches within a VCStack exchange their stack management traffic and user data over their high speed stacking links. The stack management traffic is pre-assigned to the egress queue 7. This is the highest value queue, and (in a stacked configuration) its traffic should not be shared with any user data. However, any CoS tagging of 7 applied to the incoming data will automatically be assigned to queue 7. You will therefore need to reconfigure your CoS to Queue settings to ensure that no user data is sent to queue 7.

To prevent this from happening, we recommend that you make appropriate changes to your queue settings (mappings) to reflect the stacking requirement previously described. For more information on this topic, see **"Mapping CoS tags to traffic types" on page 62.12**.

This process should include (but not be limited to) running the following command to ensure that any remaining user packets still carrying a CoS 7 tag will be mapped to egress queue 6.

To remap priority CoS traffic to egress queue 6, run the following commands:

```
awplus# config terminal
awplus(config)# mls qos map cos-queue 7 to 6
```

Stack Member Failure and Recovery

Fixed or Virtual MAC Addressing

A VCStack operates using a single virtual MAC address. This address is configurable by using the **stack virtual-mac** command on page 111.32.

Enabling the stack virtual-mac

When the **stack virtual-mac** command is enabled, the stack uses a virtual MAC address selected from an allocated pool of MAC addresses. The stack will then always use this MAC address even if the stack master fails or is removed from the stack. In this situation, the new elected master will still retain the originally configured virtual MAC address.

The virtual MAC address will be used for all external ports, and VLAN interfaces, except the management VLAN. Although each individual switch in the stack retains its own native MAC address; this is only used over the stack management VLAN.

Note If one stack member has the virtual MAC address feature enabled and another has the virtual MAC address feature disabled then they will be able to form together as a stack. From master election onwards, the stack master's virtual MAC address setting will be used by the rest of the stack.

Caution **Stack operation is only supported if stack virtual-mac is enabled.**

Virtual MAC format and value

The virtual MAC address is selected from within the range 0000.cd37.0000 to 0000.cd37.0FFF.

This can be considered as a MAC prefix component of 0000.cd37.0xxx.

where xxx is called the stack virtual-chassis-ID, and has the range 000 to FFF.

By default, the virtual-chassis-ID is randomly selected from the available range.

To change the virtual MAC address, use the **stack virtual-chassis-id command on page 111.31**.

Note Using the same virtual MAC address settings on stacks in the same network will result in duplicate MAC addresses and network disruption.
Please ensure each stack in your network uses a unique virtual-chassis-ID.

Manually selected virtual address

To manually select a virtual MAC address you enable the stack virtual MAC feature by using the commands:

```
awplus# configure terminal
awplus(config)# stack virtual-mac
```

Then configure the **stack virtual-chassis-id** command on page 111.31 to set a stack virtual-chassis-ID of your chosen value - entered as a decimal number within the range 0 to 4095. The value 120 is used in the following example:

```
awplus# configure terminal
awplus(config)# stack virtual-chassis-id 120
```

Automatically selected virtual address

If you set the **stack virtual-mac** command without entering a value for the stack virtual-chassis-ID, the switch will randomly select a virtual-chassis-ID from the allocated range.

Disabling the stack virtual-mac

When the **stack virtual-mac** command is disabled, the stack will use the MAC address of the current Master. If the stack master fails, the stack MAC address changes to reflect the new master's MAC address. If the stack MAC address does change, ARP tables of devices on the network will update to reflect the change in MAC address via **ip gratuitous-arp-link** command on page 29.27.

Stack Resiliency Link

The purpose of the resiliency link is to provide the stack members with status information that allows them to detect whether the stack master is still operational after a stack failure occurs.

Using the resiliency link, a stack member can differentiate between the master suffering a power-down or a software lock-up, where the master is offline, compared with a stacking-link failure, where the master is still online but connectivity over the stacking cables has been lost.

This enables the other stack members to either operate in the fall-back Disabled Master mode, or to re-elect a new stack master. The **"State change table" on page 110.13** shows how the stack members respond to various problems occurring on the master node.

Stack recovery states

The following state-change-table shows stack member failure conditions and recovery actions in situations where the resiliency link is present or absent.

Table 110-1: State change table

Event on Master Node	Reaction on Master	Reaction on Stack Member with resiliency link	Reaction on Stack Member without resiliency link
Both stack links removed	No change	**Disabled master**	Re-elect master [2]
Hardware reset (or fault)	Reset / offline	Re-elect master	Re-elect master [2]
Run the **no stack enable** command[3]	No change	**Disabled master**	**Disabled master**
Software application problem (lock-up or continual crashes)	Reboot as stack member	Re-elect master	Re-elect master
Software crash or lock-up	Frozen[4]	Re-elect master	Re-elect master
Power-down or PSU failure	Powered down	Re-elect master	Re-elect master

Table 110-1: State change table

Event on Stack Member Node	Reaction on Master	Reaction on Stack Member	Reaction on Stack Member
Both stack links removed	No change	**Disabled master**[1]	Re-elect master [2]
Hardware reset (or fault)	No change	Reset/offline	Reset/offline
Run the **no stack enable** command[3]	No change	**Disabled master**	**Disabled master**
Software application failover (lock-up or continual crashes)	No change	Re-boot as backup member	Reboot as stack member
software crash or lock-up	No change	Frozen[4]	Frozen[4]
Power-down / PSU failure	No change	Powered Down	Powered down

If a backup member becomes the **Disabled master** it will first disable all its switch ports, then activate any triggers specified with the **type stack disabled-master** command that have been configured.

1. The stack member assumes the role of stack master. In specific situations this condition could result in a stack containing two masters. This would present problems with network management and the control of links that were previously aggregated.
2. The following situation will apply to a switch that has been removed from the stack by the using the “no stack enable” command:
   - « The switch will be unaware of further faults within the stack it was previously a member of.
   - « Should this switch then be powered down, all links previously shut down (as a result of running the no stack enable command) will remain down.
3. If the backup member’s ports are still up, this may cause downstream switches with trunked ports to operate incorrectly.

Stack Failure Recovery

If the stack master either fails, or is removed, the other stack members will re-elect a new stack master. The stack members automatically determine which of them becomes the new stack master. See the **Disabled Master Monitoring (DMM)** and the **Disabled master** sections for more information. Alternatively, you can manually configure a trigger with the **type stack disabled-master** command on page 103.28 to activate on a stack member if it becomes the disabled master.

Table 110-1 shows how the stack backup members would respond to various problems occurring on the stack master stack.

Note

When VCStack is used with EPSR, the EPSR **failovertime** must be set to at least 5 seconds to avoid any broadcast storms during failover. Broadcast storms may occur if the switch cannot failover quickly enough before the EPSR **failovertime** expires. See the **epsr** command for further information about the EPSR **failovertime**.

See the **reboot rolling** command for further information about stack failover.

Stack Separation and Recovery

Stack stubs occur when a fault results in the stack splitting into two, with one of the stack members taking on the role of stack master. Where the stack master is still active after a fault, and other stack members are not aware that the stack master is still active, the result can be two independently operating stacks, or stubs.

When two stub stacks are reconnected, a dual master situation will be detected, and the console log will display the message that a 'duplicate master' was detected. This situation results in the re-election of the stack master based upon the lowest Priority ID, or, where both members have the same Priority ID, the lowest MAC address. The 'losing' master and other prospective stack members will then reboot and join the new stack as backup stack members.

Note Stubs are unlikely to cause network connectivity problems if a resiliency link is used.

Stack Maintenance

Adding a stack member

An unstacked switch can be added to an existing stack (hot-swapped in) with minimal impact on traffic. To do this, power down the new member switch, then connect its stacking ports and power on the switch. The switch will boot as a member of the stack.

Note The existing Stack Member-ID and the device MAC address will have no effect on the status of the new member switch. The stack will admit the new device as an ordinary stack member and allocate it a new Stack Member-ID if its ID is one that already exists.

However, for good practice we recommend pre-configuring the new member with settings that are appropriate for when the new switch becomes a stack member.

This is to avoid unexpected situations occurring when the stack is rebooted.

For example, if the new member had a priority setting that was lower than 128 and all the existing stack members were configured with the default; then, when the stack is rebooted, the new member would be elected as the stack master.

Replacing a stack member

A stack member can be removed from a stack (hot-swapped out) with minimal impact on stack traffic. To do this, power-down the stack member, and disconnect its stacking ports.

You can seamlessly swap a stack member switch into the stack to replace another with the same configuration. This provides a simple way to replace an out-of-service switch with minimal impact, and minimal administration requirement. Before inserting the replacement device into the stack, make sure that:

- The replacement device is running a compatible firmware version.
- You set the Stack ID on the replacement device to the same ID as the device being replaced.
- The replacement device is installed with the same licenses as the other stack members — this ensures consistent behavior across the stack.

Once these requirements are met, insert the new stack member, reconnect the stacking ports and power-up the new stack member.

Combining separate stacks

Two small (2 member) stacks can be combined into a single 4 member stack by physically reconnecting the stack members and rebooting. Note that the likelihood of a successful stack recombination is greatly increased if you set the stack IDs of each stack member to be unique within the combined stack that you are creating.

For example, consider two individual 2 member stacks that are to be combined into one 4 member stack, where the members of each stack had the stack member IDs of 1 and 2. Then, before you combine the stacks, you should renumber the member IDs of stack two to be 3 and 4.

Disabled master

A properly functioning VCStack contains an (active) master and one or more (backup) members. Under fault conditions it is possible for some of the stack members to lose connectivity with the stack master. In this situation the stack members without master connectivity will form themselves into a stack stub and elect a “Disabled Master” to manage the stub until the fault is rectified. Once elected to this state the disabled master will disable all of its own ports and those of all other stack members within the stub. Apart from this, the operation and ‘look and feel’ of a disabled master is very similar to an active master.

By disabling all the stub’s switchports, the disabled master avoids potential network connectivity problems that could result from by having two stack masters using the same configuration, or two switches in separated stubs trying to share the same “logical” communications paths such as a non functioning aggregated links. The active master’s ports are unaffected by this, and they will continue to forward traffic normally.

Note that status information for members of the stack stub can be accessed by logging into the disabled master, in the same way as obtaining status information for a normal stack.

Note

In a stack of more than two units, several stack members could become separated from the stack master. In this case, these stack members will form a stack stub together. Only one stack member will become the disabled master. The other switches will remain as backup stack members, but their switchports will still be disabled. Status information for the stack members can be accessed by logging into the disabled master, in the same way as obtaining status information for a normal stack.

Disabled Master Monitoring (DMM)

The stack resiliency link and disabled master state offer a unique prevention of catastrophic network connectivity problems. However, when stack members become separated, the network is still left in a fragile state where the stack master no longer has the redundancy of a backup stack member. If the original stack master were to subsequently fail whilst the stack was separated, then all network connectivity would be lost if the disabled master's switchports remained shut down.

The Disabled Master Monitoring (DMM) feature avoids this situation by continuing to monitor the status of the original stack master (the active master) via the stack resiliency link. When the DMM feature is enabled, the disabled master can detect a failure of the original stack master within a few seconds. If a failure is detected, the disabled master transitions to the active master state and automatically re-enables all its switchports. This allows traffic forwarding via the stack to continue.

For more information about the disabled master state, see the section **"Disabled master" on page 110.16**. Note that a disabled master has the same configuration as an active master, but a disabled master has all its links shut down.

Table 110-2: Stack behavior comparison

Stack behavior with DMM disabled	Stack behavior with DMM enabled
■ The VCStack breaks with the Stack Resiliency Link configured and enabled.	■ The VCStack breaks with the Stack Resiliency Link configured and enabled.
■ The separated stack member becomes a disabled master.	■ The separated stack member becomes a disabled master.
■ The disabled master does not monitor the active master.	■ The disabled master monitors the active master.
■ If the active master fails then the disabled master does not become the active master (no state transition).	■ If the active master fails then the disabled master becomes the active master (disabled to active transition).
■ No switchports are re-enabled on the disabled master. No traffic is forwarded.	■ Switchports on the disabled master are re-enabled. Traffic is still forwarded.

To enable the DMM feature, use the commands:

```
awplus# configure terminal
awplus(config)# stack disabled-master-monitoring
```

To disable the DMM feature, use the commands:

```
awplus# configure terminal
awplus(config)# no stack disabled-master-monitoring
```

To show the status of DMM on the VCStack, use the command:

```
awplus# show stack [detail]
```

To apply a trigger upon transition from active master state to disabled master state, use the command:

```
awplus# type stack disabled-master
```

To apply a trigger upon transition from disabled master state to active master state, use the command:

```
awplus# type stack master-fail
```

Note

A disabled master trigger allows you to specify a script to reconfigure the disabled master on the fly, should a catastrophic failure separate the stack. This is useful to configure an alternate IP address so you can still log in to the disabled master via an SSH or a Telnet connection. The trigger script should use the **no shutdown** command to re-enable any switchports needed for an SSH or a Telnet management connection.

Provisioning (Stack Members)

Stack member provisioning is the pre-configuration of a stack member's position ready for insertion at a later time. Provisioning enables a network administrator to pre-configure vacant stack member capacity within a VCStack, ready to be hot-swapped in at a later time. Later, when the stack member switch is physically added, its configuration is automatically applied with the minimum network disruption. Provisioning is ON by default, and cannot be disabled.

Provisioned capacity can be applied by either of the following actions:

- applying the **switch provision (stack)** command on page 111.33
- installing, then removing a provisionable device from its physical location, that is, a switch from its stack.

Provisioned Board Classes

Provisioning introduces the concept of defined board classes. **Table 110-3** lists the stack member classes that have been defined for provisioning. Each board class is assigned a class and an appropriate port count. Presently no further definitions have been made for additional features such as media type, or PoE capability. This structure simplifies configuration.

Table 110-3: Provisioned stack member classes

Board Classes	
Class	Port Count
x510-28	24 switch ports, plus 4 SFP+ ports, two of which can be used for VCStacking.
x510-52	48 switch ports, plus 4 SFP+ ports, two of which can be used for VCStacking.

Applying Hardware Provisioning

As previously mentioned, provisioning is the pre-configuration of vacant (i.e. unused) device capacity ready for device insertion at a later time.

Without provisioning

On software versions prior to version 5.3.4, trying to configure an unused port will result in the following error message:

```
awplus(config)# interface port4.0.1
% Can't find interface port4.0.1
```

With provisioning

With provisioning, you can configure stack members and their ports ready for future addition, even though they are not currently physically present:

```
awplus(config)# switch 2 provision <switch-model>
```

For information on using this command, see **"switch provision (stack)" on page 111.33**.

Now that the switch is provisioned within the stack—although not yet physically present—you can move on to provisionally configure the switch ports themselves. The following example sets the port speed of port 2.0.1 to be 1000 Mbps.

```
awplus(config)# interface port2.0.1
awplus(config-if)# speed 1000
```

You can apply provisional configuration to all interface related commands. However, you cannot apply provisioning where it changes the network's physical topology. For example, you can't provision a switch as stack member 3 and then later change it - while its position is still vacant - to stack member 4. In this situation, you would need to unprovision the switch, then provision it again as stack member 4.

The following example creates a provisioned configuration that shows the association of ports with a VLAN:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 12 state enable
awplus(config-vlan)# exit
awplus(config)# interface port2.0.1
awplus(config-if)# switchport mode access
awplus(config-if)# switchport access vlan 12
awplus(config-if)# exit
awplus(config)# interface port2.0.2-port2.0.26
awplus(config-if)# switchport mode access
```

Provisioning error messages

The following error messages may appear when configuring provisioning:

Table 110-4: Provisioning error messages (switch x [bay y] provision)

Error Message	Comment
`Switch %d must be provisioned before bays can be provisioned.`	`%d` = stack member id
`% Switch %d (%s) has 0 expansion bays.`	`%d` = stack member id `%s` = board class
`% Board class %s is incompatible with this system.`	`%s` = board class from cmd line
`% switch %d is already populated with %s.`	Indicates an attempt to provision a switch that is already present.
`% switch %d is already provisioned for %s.`	Indicates an attempt to provision a switch that is already provisioned.

Table 110-5: Provisioning error messages (no switch x [bay y] provision)

Error Message	Comment
`% switch %d bay %d is currently populated by %s`	You must remove hardware before unprovisioning

Removing Hardware Provisioning

Hardware capacity that has been previously provisioned and is presently unoccupied can be unprovisioned with the no switch provision command. This removes the provisioned configurations for hardware that has either not yet been physically added to a switch or VCStack, or has previously existed, but has been removed.

The no switch provision command will also delete any switch bay commands with the same unit number and all associated interfaces, as well as all configuration for that switch .

You cannot unprovision hardware that is currently installed. A no switch command will not succeed if the unit/unit.bay location is currently occupied. For example:

```
awplus(config)# no switch 2 bay 2 provision

                % switch 2 bay 2 is currently populated by
                <module-type>
```

The following example displays the output of a show stack that includes a provisioned VCStack member 3:

```
Virtual Chassis Stacking summary information
ID  Pending ID  MAC address      Priority  Status   Role
1   -           0000.cd28.5377   128       Ready    ActiveMaster
2   -           0000.cd29.95bf   128       Ready    BackupMember
3   -           0000.0000.0000   -         -        Provisioned
```

Switch ID 3 is then unprovisioned:

```
awplus(config)# no switch 3 provision
```

Run the show stack command to confirm that switch 3 has been unprovisioned:

```
awplus# show stack
```

```
Virtual Chassis Stacking summary information
ID  Pending ID  MAC address      Priority  Status  Role
1   -           0000.cd28.5377   128       Ready   ActiveMaster
2   -           0000.cd29.95bf   128       Ready   BackupMember
```

Note Ensure that you save your running configuration to your startup configuration after making any provisioning changes using **copy running-config startup-config.**

Displaying Provisioned Configurations

In this respect the major difference associated with provisioning, is that interface configurations will still exist in the config files and will appear in show commands, even though a device itself may not be physically installed. This (provisioning) could result from device capability that has been preconfigured for future installation, or could result from the removal of an installed device.

The show running-config command includes switch commands for existing hardware, plus all non-existent, but provisioned, hardware. The following example output of the show running-config command illustrates how provisioned and existing hardware is displayed.

Figure 110-10: Sample display of existing and provisioned show output

```
sh running-config
!
switch 1 provision x510
!
interface port1.0.1-1.0.12
 switchport
 switchport mode access
!
.end
```

Displaying provisioned hardware status

The status, present or provisioned, appears in monitoring commands such as the show interface brief command. The following sample output from the show interface brief command shows the provisioning status of two configured stack members.

Figure 110-11: Sample show interface brief output showing hardware provisioning status

```
awplus#show interface brief
Interface              Status          Protocol
port2.0.24             admin up        down
port3.0.1              admin up        provisioned
port3.0.2              admin up        provisioned
```

A more detailed inspection of the provisioned port2.0.3 is shown below. Note that the MAC address is 0000.0000.0000, which is the value applied as a placeholder for all provisioned ports. Also note that although the port is in the link DOWN state its administrative state of UP PROVISIONED means that it can be further configured. For example, it can be associated with a VLAN, or added to a link aggregation group.

Figure 110-12: Sample display showing provisioning status of a specific port

```
Interface port2.0.3
  Scope: both
  Link is DOWN, administrative state is UP PROVISIONED
  Thrash-limiting
    Status Unknown, Action learn-disable, Timeout 1(s)
  Hardware is Ethernet, address is 0000.0000.0000
  index 6801 metric 1 mtu 1500 mru 1522
  <BROADCAST,MULTICAST>
  SNMP link-status traps: Disabled
    input packets 0, bytes 0, dropped 0, multicast packets 0
    output packets 0, bytes 0, multicast packets 0 broadcast
pks0
```

Provisioning and Configuration Management

A benefit of provisioning is configuration settings are no longer dependent on the existence of hardware devices. When a device is removed, all the interfaces for that device are shut down and its provisioning status is set. This means that you can add or remove physical hardware without affecting your network interfaces. Of course when ports go down (i.e. are physically removed) there will be other changes to network configuration, as protocols may re-converge or routes may be removed.

Switches within a VCStack can be hot-swapped without the need for reconfiguration.

The configuration of a newly inserted device that matches the provisioned board-class is achieved on a best-effort basis. For example inserting a non-POE switch into a stack member location configured for PoE will result in the failure of the PoE configuration commands.

Take care that your provisioned configurations, match with the type of hardware that you plan to install.

```
awplus(config)# switch 2 bay 4 provision <product-type>
awplus(config)# interface port2.0.4
awplus(config-if)# shutdown
```

Software Version Auto Synchronization

Introduction

Different software releases exhibit functional and operational differences. To maintain consistent behavior across the stack, all new member switches must be running the same software release before they can fully join the stack.

Manually upgrading the software release of each new stack member that joins a stack would be a cumbersome process. The VCStack software version auto synchronization feature automates this process by ensuring the same software release is used on all stack members, and automatically upgrades stack members where required.

Note that to maintain consistent behavior across the stack, all member switches should have the same licenses enabled. See the **Stack License Management** section in this chapter and see **Chapter 8, Licensing Introduction and Configuration** for an overview of feature licensing that should be applied to each stack member. Stack members without feature licensing applied operate in an unlicensed, unsupported mode. To purchase feature licenses, contact your authorized distributor or reseller.

How Auto Synchronization Works

Software version comparison

When the stack is formed, it elects one of its switches to become the master. The software release running on the stack master will then become the software version used throughout the stack. After a master is elected, all the stack members compare their current software version with the version that is running on the stack master.

If the comparison process detects differences between software versions, the software version auto synchronization feature will automatically copy the master's software release onto the appropriate stack members. Once the software release has successfully been copied, this version will become the boot software for that particular stack member, which will then reboot in order to load the new software release.

If a software version running on a stack member is incompatible with that running on the master, and software version auto synchronization is turned off, then that switch will be removed as a stack member. See **stack software-auto-synchronize command on page 111.30**.

When auto-synchronization upgrades a stack member, the member's current running software will be set as the backup software release. If there are any problems loading the new software, then the backup software release will be used to recover.

If the stack member does not have enough free Flash memory space for the new release, then the new release will replace up to two older release files in Flash memory, which is determined by software build dates of the older release files. The oldest release files are replaced first.

Auto synchronization limitations

Because the stack master's software version gets applied to the rest of the stack, care must be taken to ensure the correct switch is elected master. If the master is running an older software release, then software version auto synchronization may downgrade the software releases running on other stack members. For configuring which stack member becomes the master, see the **stack priority command on page 111.25**.

Software version auto synchronization will not work if stack members are booting using either a one-off boot or from TFTP or YMODEM. In these situations, any stack members running different software will boot as standalone devices.

If software version auto synchronization is configured as **off** for a stack member that is running a different software release to the master, by applying the command "no stack software-auto-synchronize" that switch will boot as a standalone device. For more information, see the **stack software-auto-synchronize command on page 111.30**.

Incompatible software releases

The software version auto synchronization feature will not always work if there have been significant VCStack or system changes between the two different software releases. The VCStack discovery of other stack members uses an internal "stack S/W version" to detect compatibility between builds.

If the VCStack software between two stack members is incompatible, the software version auto synchronization feature will not work. Instead, an "incompatible stack S/W version" log message will be displayed and both stack members will boot as standalone devices. This is an undesirable situation because both devices may load the same configuration file, which could cause network conflicts. In order to avoid this situation when upgrading the stack to a new major release, ensure the "boot system" command succeeds.

In general, the software version auto synchronization feature will work between maintenance releases, such as between 5.4.4-0.1 and 5.4.4-0.2, but may not work between major or minor releases that have new VCStack features, such as between 5.4.3-0.1 and 5.4.4-0.1. For information about which specific releases support auto synchronization, see the **New and Enhanced Features** document for the release.

Upgrading stack software reliably

When upgrading a stack to a new software release, the **boot system backup** command on page 7.11 will automatically synchronize the new software release across stack members. If there is insufficient file space on a backup member, the boot system command has an interactive mode that prompts you to delete old releases to free up file space.

However, if you choose not to delete any release files, or if Flash space is taken up with other types of files, then the boot system command can fail to set the preferred release on the backup member. If you do not have enough space in Flash to add a backup release file, then remove unused files in Flash using the delete command.

If you are unsure which files to delete, the following process may assist you.

```
awplus# remote-login 2
awplus-2# enable
awplus-2# dir
```

Use the remote-login command to login to the backup member with insufficient free file space, in this example member 2.

Look for any .rel (release) files, .jar (GUI), or .tgz and .gz (diagnostic) files that are no longer needed and use the delete command to remove them.

Alternatively, you can use the file system commands directly from the master's console prompt using the filepath of the backup member's Flash. Substitute `awplus` for the hostname in the configuration, and use `awplus-2` for stack member 2, and so on.

```
awplus# dir awplus-2/flash:*

awplus# delete awplus-2/flash:/x510-5.4.4-
        3.5.rel
```

Stack License Management

Feature licenses across a stack can be managed either via the CLI with the **license** command or via SNMP using the AT-LICENSE-MIB Enterprise MIB.

See **Chapter 8, Licensing Introduction and Configuration** for an introduction and overview of licensing and configuration tasks. See **Chapter 9, Licensing Commands** for command descriptions and examples.

Stack members without licensing applied operate in an unlicensed, unsupported mode. To purchase feature licenses, contact your authorized distributor or reseller.

The stack does not have to be restarted when a feature license is enabled or disabled. When a new feature license is enabled across the stack some protocol modules may need to be restarted before the license is activated. A warning is given about the need to restart the protocol modules and a "y/n" prompt is given to proceed with the license installation. If SNMP is used to enable a feature license, the affected protocol modules are restarted automatically.

To maintain consistent behavior across the stack, all member switches should have the same feature licenses enabled. However, you can enable or disable a feature license on a single stack member if required. Note that doing so could result in the stack failing to operate correctly. If you enable or disable a feature license on a single stack member via the CLI a warning message is generated. A warning message is not generated if you manage the stack via SNMP.

The license keys available for purchase include a limit on the number of switches that it can be applied to. If the license is applied to a stack that has more members than the license is valid for via the CLI, a warning message is generated and the event is logged.

For feature licenses, contact your authorized distributor or reseller. If a license key expires or a proper key is not installed, some software features will not be available.

Note

See the AlliedWare Plus™ datasheet for a list of current feature licenses available by product, and the AlliedWare Plus™ How To notes for information on obtaining them.

To enable a feature license on all stack members, use the command:

```
awplus# license <label> <key>
```

To disable a feature license on all stack members, use the command:

```
awplus# no license <label>
```

To display detailed information about feature licenses on the stack master, use the command:

```
awplus# show license [feature] [<label>|index <index-
        number>]
```

To display detailed information about feature licenses on a single stack member or all stack members, use the command:

```
awplus# show license [<label>] member [<1-8>|all]
```

To display brief information about feature licenses on a single stack member or all stack members, use the command:

```
awplus# show license [<label>] brief member [<1-8>|all]
```

Chapter 111: Stacking Commands

Introduction 111.2
Command List 111.3
clear counter stack 111.3
debug stack 111.4
reboot rolling 111.5
reload rolling 111.5
remote-command (deleted) 111.6
remote-login 111.7
show counter stack 111.8
show debugging stack 111.12
show running-config stack 111.13
show provisioning (stack) 111.14
show stack 111.15
show stack resiliencylink 111.20
stack disabled-master-monitoring 111.21
stack enable 111.22
stack management subnet 111.23
stack management vlan 111.24
stack priority 111.25
stack renumber 111.26
stack renumber cascade 111.27
stack resiliencylink 111.28
stack software-auto-synchronize 111.30
stack virtual-chassis-id 111.31
stack virtual-mac 111.32
switch provision (stack) 111.33
switchport resiliencylink 111.34
vlan mode stack-local-vlan 111.35
undebug stack 111.36

Introduction

This chapter provides an alphabetical reference for each of the Stacking commands.

Also note the following stacking trigger commands that are documented in the Triggers chapter:

type stack disabled-master command on page 103.28

type stack master-fail command on page 103.29

type stack member command on page 103.30

type stack link command on page 103.29

In addition to the stacking commands shown in this chapter, stacking content also exists in the following commands:

hostname command on page 10.18

reboot command on page 10.23

reload command on page 10.23

show cpu command on page 10.26

show cpu history command on page 10.29

show exception log command on page 12.39

show file systems command on page 7.36

show memory command on page 10.36

show memory history command on page 10.39

show process command on page 10.43

show system command on page 10.47

Caution

Stack operation is only supported if stack virtual-mac is enabled. For more information refer to: "Enabling the stack virtual-mac" on page 110.12 and "stack virtual-mac" on page 111.32

Command List

clear counter stack

This command clears all stack counters for all stack members.

Syntax `clear counter stack`

Mode Privileged Exec

Example To clear all stack counters:

```
awplus# clear counter stack
```

Related Commands show counter stack

debug stack

This command enables the stacking debugging facilities.

Syntax

```
debug stack [link|topology|trace]
no debug stack [link|topology|trace]
```

Parameter	Description
`link`	Stacking neighbor discovery events on stack links.
`topology`	Stacking topology discovery messages.
`trace`	Notable stacking events.

Default Stack trace debugging is enabled.

Mode Privileged Exec and Global Configuration

Usage The command displays debug information about the stacked devices. If no parameter is specified, all the stack debugging information will be displayed, including link events, topology discovery messages and all notable stacking events. If link parameter is specified, only the link events debugging information will be displayed.

Examples To enable debugging, enter the following command on the stack master:

```
awplus# debug stack
```

To enable link debugging, enter the following command on the stack master:

```
awplus# debug stack link
```

To enable topology discovery debugging, enter the following command on the stack master:

```
awplus# debug stack topology
```

To enable stack trace debugging, enter the following command on the stack master:

```
awplus# debug stack trace
```

Related Commands undebug stack

reboot rolling

This command reboots a stack in a rolling sequence to minimize downtime.

The stack master is rebooted, causing the remaining stack members to failover and elect a new master. The rebooted unit remains separate from the remaining stack and boots up as a stand-alone unit. Once the rebooted unit has finished running its configuration and has brought its ports up, it reboots all the remaining stack members at once.

Syntax `reboot rolling`

Mode Privileged Exec

Usage If you are upgrading to a new software version, the new version must also support rollling reboot.

Note When stacking is used with EPSR, the EPSR **failovertime** must be set to at least 5 seconds to avoid any broadcast storms during failover. Broadcast storms may occur if the switch cannot failover quickly enough before the EPSR **failovertime** expires. For further information about EPSR **failovertime**, see the **epsr** command on page 84.4.

Examples To rolling reboot the stack, use the following commands:

```
awplus# reboot rolling
Continue the rolling reboot of the stack? (y/n):
```

After running this command, the stack master will reboot immediately with the configuration file settings. The remaining stack members will then reboot once the master has finished re-configuring.

```
Continue the rolling reboot of the stack? (y/n):
awplus# y
```

Related Commands **boot system**
epsr

reload rolling

This command performs the same function as the **reboot rolling** command on page 111.5.

remote-command (deleted)

This command has been deprecated. Please use the **remote-login** command on page 111.7, and then the following commands:

show cpu command on page 10.26

show counter stack command on page 111.8

show exception log command on page 12.39

show file command on page 7.35

show log permanent command on page 12.45

show memory command on page 10.36

show process command on page 10.43

show stack command on page 111.15

show system command on page 10.47

clear counter stack command on page 111.3

This command executes a number of host-directed commands that are specific to stack members.

remote-login

This command is used only on the master in order to log onto the CLI of another stack member. In most respects the result of this is similar to being logged into the stack master. Configuration commands are still applied to all stack members, but show commands, and commands that access the file system are executed locally.

The specific output obtained will vary greatly depending on the show command chosen.

Syntax `remote-login <stack-ID>`

Parameter	Description
`<stack-ID>`	Stack member number, from 1 to 4.

Mode Privileged Exec

Usage Note that some commands such as **ping** or **telnet** are not available when the remote-login is used.

Example To log onto stack member 2, use the following command:

```
awplus# remote-login 2
```

To return to the command prompt on the master stack member, type **exit**.

show counter stack

Use this command to display stack related counter information.

Syntax `show counter stack`

Default All counters are reset when the stack member is rebooted.

Mode User Exec and Privileged Exec

Usage This displays the stacking counter information for every stack member.

Examples To display the stacking counter information about the whole stack, use the following command.

```
awplus# show counter stack
```

Figure 111-1: Example output from the show counter stack command

```
Virtual Chassis Stacking counters

Stack member 1:

Topology Event counters
Units joined          ......... 1
Units left            ......... 0
Links up              ......... 1
Links down            ......... 0
ID conflict           ......... 0
Master conflict       ......... 0
Master failover       ......... 0
Master elected        ......... 1
Master discovered     ......... 0
SW autoupgrades       ......... 0

Stack Port 1 Topology Event counters
Link up               ......... 3
Link down             ......... 2
Nbr re-init           ......... 0
Nbr incompatible      ......... 0
Nbr 2way comms        ......... 1
Nbr full comms        ......... 1

Stack Port 2 Topology Event counters
Link up               ......... 0
Link down             ......... 0
Nbr re-init           ......... 0
Nbr incompatible      ......... 0
Nbr 2way comms        ......... 0
Nbr full comms        ......... 0
```

Figure 111-1: Example output from the show counter stack command (cont.)

```
Topology Message counters
Tx Total                ......... 4
Tx Hellos               ......... 4
Tx Topo DB              ......... 0
Tx Topo update          ......... 0
Tx Link event           ......... 0
Tx Reinitialise         ......... 0
Tx Port 1               ......... 4
Tx Port 2               ......... 0
Tx 1-hop transport      ......... 4
Tx Layer-2 transport    ......... 0
Rx Total                ......... 1
Rx Hellos               ......... 1
Rx Topo DB              ......... 0
Rx Topo update          ......... 0
Rx Link event           ......... 0
Rx Reinitialise         ......... 0
Rx Port 1               ......... 1
Rx Port 2               ......... 0
Rx 1-hop transport      ......... 1
Rx Layer-2 transport    ......... 0

Topology Error counters
Version unsupported     ......... 0
Product unsupported     ......... 0
XEM unsupported         ......... 0
Too many units          ......... 0
Invalid messages        ......... 0

Resiliency Link counters
Health status good      ......... 1
Health status bad       ......... 0
Tx                      ......... 0
Tx Error                ......... 0
Rx                      ...... 3600
Rx Error                ......... 0

Stack member 2:

-- Output repeated for other stack members - details not shown--
```

Table 111-1: Parameters in the output of the show counter stack command

Parameters	Description
`Topology Event Counters`	
`Units joined`	Number of times that the stack acquires a member.
`Units left`	Number of times that the stack loses a member.
`Links up`	Number of times that a stack link is up in the stack.
`Links down`	Number of times that a stack link is down in the stack.
`ID conflict`	Number of times that stack-ID conflicts.
`Master conflict`	Number of times that stack master conflict occurs.
`Master failover`	Number of times that stack master fails.
`Master elected`	Number of times that stack master is elected.
`Master discovered`	Number of times that stack master is discovered.
`SW autoupgrades`	Number of times that the software in the stack members are auto upgraded.
`Stack port counters`	

Table 111-1: Parameters in the output of the show counter stack command (cont.)

Parameters	Description
`Link up`	Number of times that this unit's physical stack link has come up.
`Link down`	Number of times that this unit's physical stack link has come down.
`Nbr re-init`	Number of times that the neighbor is detected as having reinitialised.
`Nbr incompatible`	Number of times that the neighbor is detected as incompatible.
`Nbr 2way comms`	Number of times that the neighbor is in two way communication status.
`Nbr full comms`	Number of times that the neighbor is in full communication status.
`Topology message counters`	
`Total`	Total number of topology messages.
`Hellos`	Number of hello messages.
`Topology DB`	Number of topology database messages.
`Topology update`	Number of topology database update messages.
`Link event`	Number of link event messages.
`Reinitialise`	Number of reinitialise messages.
`1-hop transport`	Number of 1-hop transport messages.
`Layer-2 transport`	Number of layer 2 transport messages.
`Link event`	Number of link event messages.
`Reinitialise`	Number of reinitialise messages.
`1-hop transport`	Number of 1-hop transport messages.
`Layer-2 transport`	Number of Layer 2 transport messages.
`Topology error counters`	Reasons why a neighboring unit could not join the stack.
`Version unsupported`	Number of stack software version unsupported errors.
`Product unsupported`	Number of product unsupported errors.
`XEM unsupported`	Number of XEM unsupported errors.
`Too many units`	Number of too many units errors.

Table 111-1: Parameters in the output of the show counter stack command (cont.)

Parameters	Description
`Invalid messages`	Number of invalid messages.
`Health status good`	The number of times that the resiliency link has successfully carried healthchecks following a failure at startup.
`Health status bad`	The number of times that the resiliency link healthcheck has timed out. A timeout occurs when a backup stack member detects a delay greater than two seconds between healthcheck messages received.
`Rx`	The total number of healthcheck messages that a stack member has received from the stack master.
`Rx Error`	The total number of invalid healthcheck messages that have been received from the master. This message is not applicable to the stack master.

Related Commands

show stack
switch provision (stack)

show debugging stack

This command shows which debugging modes are currently enabled for stacking.

Syntax `show debugging stack`

Mode User Exec and Privileged Exec

Example To display the stack debugging mode status, use the command:

```
awplus# show debugging stack
```

Figure 111-2: Example output from the show debugging stack command

```
Virtual Chassis Stacking debugging status:
  VCS link debugging is on
  VCS topology debugging is on
  VCS trace debugging is on
```

Related Commands debug stack

show running-config stack

Use this command to display the running system information specific to the stack.

```
show running-config stack
```

Mode Privileged Exec and Global Configuration

Example To display the stacking running configuration information, use the command:

```
awplus# show running-config stack
```

Output **Figure 111-3: Example output from the show running-config stack command**

```
awplus#show running-config stack

stack virtual-mac
stack virtual-chassis-id 1982
stack management vlan 4000
stack management subnet 192.168.254.0
stack enable
stack 2 priority 20
```

Related Commands **show running-config**

show provisioning (stack)

Use this command to display the provisioning status of all installed or provisioned hardware. Provisioning is the preconfiguration necessary to accommodate future connection of hardware items such as a switch.

Syntax `show provisioning`

Mode User Exec and Privileged Exec

Example To show provisioning, use the following command:

```
awplus# show provisioning
```

Output **Figure 111-4: Example output from the show provisioning command**

```
Switch provisioning summary information

ID  Board class Status
1.0 x510-28     Hardware present
```

Table 111-2: Parameters in the output of the show provisioning command

Parameter	Description
`ID`	The unit bay-location of the hardware provision.
`Board class`	The hardware type.
`Status`	The provisioned state: ■ Hardware Present means that the hardware is currently installed in the stack. ■ Provisioned means that although the hardware is not currently installed, the stack is preconfigured ready to accept the hardware installation.

show stack

Use this command to display information about current stack members.

Syntax `show stack [detail]`

Parameter	Description
`detail`	Display detailed stacking information.

Default Display summary information only.

Mode User Exec and Privileged Exec

Usage This command displays information about current stack members. If the **detail** parameter is specified, additional information will be displayed for each stack member. By default, only summary information is displayed.

Example To display summary information about the stack, use the command:

```
awplus# show stack
```

Output **Figure 111-5: Example output from the show stack command**

```
Virtual Chassis Stacking summary information

ID  Pending ID  MAC address      Priority  Status  Role
1   -           0000.cd28.07e1   128       Ready   Active Master
2   -           0015.77c2.4d44   128       Ready   Backup Member
3   -           0015.77c9.7464   128       Syncing Backup Member
4   -           -                -         -       Provisioned

Operational Status                Normal operation
Stack MAC address                 0000.cd28.07e1
```

Table 111-3: Parameters in the output from the show stack command

Parameter	Description
`ID`	Stack-ID.
`MAC address`	Stack member MAC address.
`Priority`	Stack member master election priority (between 0 and 255). Note that the lowest number has the highest priority.
`Role`	Stack member's role in the stack, this can be one of: ■ **Active Master** ■ **Disabled Master** — this is the temporary master when there is a communication break within the stack, but communication still exists across the resiliency link. In this state all switch ports within the stack are disabled by default, but a different configuration can be run by a "type stack disabled-master" trigger. ■ **Backup Member** — a device other than the stack master. ■ **Provisioned** — indicates that the stack position is provisionally configured, i.e. ready to accept a particular switch type into the stack.

Example To display the detailed stacking information about the stack's overall status:

```
awplus# show stack detail
```

Figure 111-6: Example output from the show stack detail command

```
Virtual Chassis Stacking detailed information

Stack Status:
-----------------------------------------------------------------
Operational Status                      Normal operation
Management VLAN ID                      4094
Management VLAN subnet address          192.168.255.0
Virtual Chassis ID                      388 (0x184)
Virtual MAC address                     0000.cd37.0184
Mixed mode                              Disabled
Disabled Master Monitoring              Enabled

Stack member 1:
-----------------------------------------------------------------
ID                                      1
Pending ID                              -
MAC address                             0000.cd28.070d
Last role change                        Wed May  7 22:31:58 2013
Product type                            x510-52GTX
Role                                    Active Master
Priority                                1
Host name                               awplus
S/W version auto synchronizaion         On
Resiliency link status                  Configured
Stack port 1.0.51 status                learned neighbor 2
Stack port 1.0.52 status                learned neighbor 3

Stack member 2:
-----------------------------------------------------------------
ID                                      2
Pending ID                              -
MAC address                             0000.cd29.716d
Last role change                        Wed May  7 23:47:21 2013
Product type                            x510-52GTX
Role                                    Backup Member
Status                                  Ready
Priority                                2
Host name                               awplus-2
S/W version auto synchronization        On
Resiliency link status                  Successful
Stack port 2.0.51 status                learned neighbor 3
Stack port 2.0.52 status                learned neighbor 1

Stack member 3:
-----------------------------------------------------------------
ID                                      3
Pending ID                              -
MAC address                             0015.77c2.4d9d
Last role change                        Wed May  7 22:31:58 2013
Product type                            x510-52GTX
Role                                    Backup Member
Priority                                3
Host name                               awplus-3
S/W version auto synchronizaion         On
Resiliency link status                  Successful
Stack port 3.0.51 status                learned neighbor 1
Stack port 3.0.52 status                learned neighbor 2
```

Table 111-4: Parameters in the output from the show stack detail command

Parameter	Description
`S/W version auto synchronization`	Whether the software-auto-synchronization feature is turned on or off.
`Host name`	The host name of the stack member.
`ID`	Stack-ID.
`Last Role Change`	The date and time the stack member last changed its role in the stack.
`MAC address`	Stack member MAC address.
`Management VLAN ID`	The VLAN ID currently used for stack management: the default is 4094.
`Management VLAN subnet address`	The current stacking management VLAN subnet address.
`Virtual Chassis ID`	The Virtual Chassis ID determines the last 12 bits of the Virtual MAC address: `0000.cd37.0xxx`
`Virtual MAC Address`	The Virtual MAC address of the stack.
`Disabled Master Monitoring`	The current Disabled Master Monitoring status. This can be: ■ **Enabled** ■ **Disabled** ■ **Inactive**
`Operational Status`	The status of the stack. This can be: ■ **Normal operation**: If any other status is displayed, it may warrant further investigation. ■ **Stacking hardware disabled:** The **stack enable** command needs to be added to the configuration to activate the stacking feature. ■ **Operating in failover mode**: This stack member has become separated from the rest of the stack, or it failed to join the stack correctly. ■ **Standalone unit**: Stacking is enabled, but no other stack members are present. ■ **Not all stack ports are up**: One or more stacking ports may be down, or stacking discovery may not have detected the neighbor successfully.
`Stack Status`	The stack's overall status. Note that a warning is issued if the stack is not connected in a standard ring topology.
`Pending ID`	The pending stack member ID. This can be changed by the **stack renumber** command on page 111.26. If there is no pending ID, the "–" symbol will display.

Table 111-4: Parameters in the output from the show stack detail command (cont.)

Parameter	Description
`Stack port status`	The status of the stack port. This can be: ■ **Down** ■ **Neighbor incompatible** ■ **Discovering neighbor** ■ **learned neighbor**
`Priority`	Stack member master election priority (between 1 and 255) Note that the lowest number has the highest priority.
`Product Type`	Stack member product type. For example, x510-28GPX.
`Provisioned`	Indicates that the stack position is provisionally configured, i.e. ready to accept a particular switch type into the stack.
`Resiliency link status`	The current status of the resiliency link. The status can be one of: ■ **Not configured** (Master or Member). ■ **Configured** (Master only). ■ **Successful**: Successfully receiving healthchecks from the Active Master. ■ **Failed** (Member only): Not receiving any healthchecks from the Active Master. ■ **Stopped**: The resiliency link is configured, but is inactive. This may occur in a Disabled Master stack, for example if the Disabled Master Monitoring feature is not used.
`Role`	Stack member's role in the stack, this can be one of: ■ **Active Master**. ■ **Disabled Master** — The temporary master when there is a communication break within the stack, but communication still exists across the resiliency link. In this state all switch ports within the stack are disabled by default, but a different configuration can be run by a "**type stack disabled-master**" trigger command. ■ **Backup Member** — a device other than the stack master. ■ **Discovering** — joining the stack.
`Status`	Indicates how readily a stack member can take over as master if the current stack master were to fail. ■ **Init** — the stack member is completing the startup initialization. ■ **Syncing** — the stack member is synchronizing state information with the stack master following startup. ■ **Ready** — the stack member is fully synchronized with the current master and is ready to take over immediately.

Related Commands

show counter stack
show stack resiliencylink
stack disabled-master-monitoring
stack resiliencylink
stack software-auto-synchronize

show stack resiliencylink

Use this command to display information about the current status of the resiliency-link across the members of the stack.

Syntax show stack resiliencylink

Mode User Exec and Privileged Exec

Example To display information about the current status of the resiliency-link across the stack members, use the command:

```
awplus# show stack resiliencylink
```

Output **Figure 111-7: Example output from the show stack resiliencylink command**

```
awplus(config)# show stack resiliencylink
Stack member 1:
-------------------------------------------------------------
Status                          Configured
Interface                       vlan4093
Interface state                 UP
Resiliency-link port(s)         port1.2.11

Stack member 2:
-------------------------------------------------------------
Status                          Successful
Interface                       vlan4093
Interface state                 UP
Resiliency-link port(s)         port2.2.11
```

Table 111-5: Parameters in the output of the show stack resiliencylink command

Parameter	Description
`Status`	The current status of the stack member's resiliency link. Can be one of: ■ **Not configured** (Master or Member). ■ **Configured** (Master only). ■ **Successful**: Successfully receiving healthchecks from the Active Master. ■ **Failed** (Member only): Not receiving any healthchecks from the Active Master. ■ **Stopped**: The resiliency link is configured, but is inactive. This may occur in a Disabled Master stack, for example if the Disabled Master Monitoring feature is not used.
`Interface`	The name of the VLAN interface that is connected to the resiliency link.
`Interface state`	The current status of the interface. Can be either up or down.
`Resiliency-link port(s)`	The switch port(s) the resiliency link is connected to.

Related Commands
- **switch provision (stack)**
- **show stack**
- **stack resiliencylink**
- **switchport resiliencylink**

stack disabled-master-monitoring

This command enables the Disabled Master Monitoring (DMM) feature. If a stack member becomes a disabled master, the DMM feature will use the stack resiliency link to continue monitoring the health of the separated stack master.

Use the **no** variant of this command to disable the DMM feature.

Syntax

```
stack disabled-master-monitoring
no stack disabled-master-monitoring
```

Default By default, Disabled Master Monitoring is enabled. However, it only operates if there is a resiliency link.

Mode Global Configuration

Usage This command enables additional stack resiliency link functionality, which is used if a stack separation occurs. For DMM to operate, a resiliency link must also be configured (**stack resiliencylink** command on page 111.28). A stack separation could result in a stack member becoming a disabled master, which has the configuration as a normal stack master except that all its switchports are shutdown.

For more information about the disabled master state, see **"Disabled master" on page 110.16**.

When the DMM feature is enabled, the disabled master will continue to monitor the health of the original stack master over the stack resiliency link connection. If the original stack master were to fail, when the DMM feature is enabled, then the disabled master will detect this and will automatically re-enable its switchports. This ensures that the stack will continue to pass network traffic, even if a catastrophic stack failure occurs.

For more information about the DMM feature when the stack member is a disabled master, see **"Disabled Master Monitoring (DMM)" on page 110.17**.

Examples To enable the DMM feature, use the following commands:

```
awplus# configure terminal
awplus(config)# stack disabled-master-monitoring
```

To disable the DMM feature, use the following commands:

```
awplus# configure terminal
awplus(config)# no stack disabled-master-monitoring
```

Related Commands
- **switch provision (stack)**
- **show stack**
- **stack resiliencylink**
- **type stack disabled-master**
- **type stack master-fail**

stack enable

This command is used on a stackable stand-alone switch to manually turn on the VCStack feature.

This command can also be run on a switch that has previously been removed from a stack (by using the **no** variant of this command) and return it to stacking operation.

The **no** variant of this command removes a selected stack member switch, as specified by the *<stack-ID>* selection in the command syntax, from the virtual chassis stack.

Syntax

```
stack enable
no stack <stack-ID> enable
```

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.

Default The VCStack feature is enabled by default. The feature automatically starts when hardware is present.

Mode Global Configuration

Usage When stack enable is entered, the **stack virtual-mac** is automatically enabled. Using virtual-mac is required in order to minimize disruption on failover.

Running the **no** variant of this command will remove the selected stack member from the stack. At this point the removed member will act as a stand-alone master and will disable all of its ports. The switch can then only be accessed via its console port. If the command is run on the master then all current members of the stack will be disabled.

To return the switch to stack membership, first connect to the switch via its console port, then run the **stack enable** command. Then save the configuration and run the **reboot** command on page 10.23. This will reboot the switch and it will re-join the stack as an ordinary member.

If the switch was previously the stack master, you might want to return it to its original stack master status. To do this you must run the **reboot** command on page 10.23 again. This time—because the switch is now a stack member—the command will reboot the whole stack and result in a new master election.

Note the following condition of applying the **no stack** *<stack-ID>* **enable** command:

- If the specified `stack-ID` is not used by any current stack member, the command will be rejected.

Caution Disabling a stack member can significantly degrade the throughput capability of the stack.

Example To turn on stacking on a stackable stand-alone unit, use the command:

```
awplus# configure terminal
awplus(config)# stack enable
```

Related Commands **reboot**

stack management subnet

This command configures the subnet address used by the stack management VLAN.

Use the **no** variant of this command to reset the stack's VLAN subnet management address back to the default address and mask (192.168.255.0/27).

Syntax

```
stack management subnet <ip-address>
no stack management subnet
```

Parameter	Description
<ip-address>	The new subnet address for the stack management VLAN.

Default The default stacking management VLAN subnet address is 192.168.255.0 with a subnet mask 255.255.255.224 or /27.

Mode Global Configuration

Usage This command configures the stack management VLAN subnet address.

The management VLAN will be used for high speed communication between stacked units via the stacking ports. Although this command enables you to change the IP address command, the subnet mask must always remain as shown.

The stack management IP subnet is solely used internally to the stacked devices, and cannot be reached external to the stack. You should only change the stack management VLAN subnet address if it causes a conflict within your network.

Note that several separate stacks can use the same default management VLAN subnet address even though their user ports may share the same external network. If the stack subnet address is changed, then the configuration for any new units must also be updated before they are inserted into the stack.

If the management VLAN subnet address is changed by this command, you can use the **no** variant of this command to reset it to it's default.

Example To set the management VLAN subnet address to 192.168.255.144:

```
awplus# configure terminal
awplus(config)# stack management subnet 192.168.255.144
```

Related Commands stack management vlan

stack management vlan

Use this command to configure the stack management VLAN ID.

Use the **no** variant of this command to change the stack management VLAN ID back to the default (VLAN ID 4094).

Syntax

```
stack management vlan <2-4094>
no stack management vlan
```

Parameter	Description
<2-4094>	Stack management VLAN ID.

Default VLAN ID 4094

Mode Global Configuration

Usage The management VLAN is used for high speed communication between stacked units. This command enables you to change the ID of this VLAN.

The default stacking management VLAN ID is 4094, which is the last configurable VLAN ID in the switch.

The stack management VLAN is created and configured automatically so that the stack VLAN cannot be used in the stack's VLAN configuration commands (such as `awplus(config-vlan)# vlan <Stack management VLAN ID>`).

The management VLAN should only be changed if the default stack VLAN ID needs to be used in the stack's VLAN configuration.

Caution

When the command is entered, the updated management VLAN configuration will take effect once the stack is restarted.

If the management VLAN ID is changed by this command, you can use the no variant of this command to change it back to default value.

Examples To set the management VLAN to 4000, enter the following commands:

```
awplus# configure terminal
awplus(config)# stack management vlan 4000
```

To reset the management VLAN back to the default (4094), enter the following commands:

```
awplus# configure terminal
awplus(config)# no stack management vlan
```

Related Commands stack management subnet

stack priority

Use this command to change a specific stack member's master-election priority.

Syntax

```
stack <stack-ID> priority <0-255>
no stack <stack-ID> priority
```

Parameter	Description
<stack-ID>	Stack member number, from 1 to 4.
priority	The stack member's election priority value.
<0-255>	The stack member's new priority value. The lowest value is assigned the highest priority. The default is 128.

Mode Global Configuration

Usage This command is used to change the value of a specific stack member's master-election priority. If the specified stack-ID is not used by any current stack member, the command will be rejected.

The election criteria selects the stack member with the lowest priority value to become the stack master. Where two stack members both have the same lowest priority value, then the stack member with the lowest MAC address will be elected as master.

Note

Assigning a new priority value will not immediately change the current stack master. In order to force a master re-election after the new priority value is assigned, use reboot stack-member <master's ID> to reboot the current stack master, a new stack master will then be elected based on the new priority values.

Example To change the priority of stack member 2 to be 3, use the command:

```
awplus# configure terminal
awplus(config)# stack 2 priority 3
```

Validation Command show stack

stack renumber

Use this command to renumber a specific stack member.

Syntax `stack <existing stack-ID> renumber <new stack-ID>`

Parameter	Description
<existing stack-ID>	We recommend that you use only numbers 1 to 2 for a 2 unit stack, or 1 to 4 for a four unit stack.
`renumber`	Change the existing `stack-ID`.
<new stack-ID>	We recommend that you use only numbers 1 to 2 for a 2 unit stack, or 1 to 4 for a four unit stack.

Default Every stack unit will initially try to use a `stack-ID` of 1.

Mode Global Configuration

Usage This command is used to change the ID of a specific stack member - primarily when exchanging stack members. The changes made by this command will not take effect until the switch is rebooted.

Note

This command does not alter any of the stacks's existing configuration, apart from the `stack-ID` specified. For example, if stack member 2 were removed from the stack and a new stack unit is assigned the member 2 `stack-ID`, then the interface configuration that existed for the removed stack member 2 will be applied to the new stack member 2.

The ***existing stack-ID*** must already be assigned to an existing stack member. To avoid duplicating IDs, a warning message will appear if you assign a ***new stack-ID*** that is currently assigned to another stack member. However, you can continue to renumber the stack-IDs and remove ID duplications. If you do not remove the duplications, then one of the devices will be forced to automatically renumber to an unused ID. Once you have removed any duplicate IDs, you can reboot the switch to implement your changes.

Note that the configured `stack-ID` is saved immediately on the renumbered member, and so is not reliant on using the **copy running-config** command for it to take effect.

Example To renumber stack 1 to stack 2, use the commands:

```
awplus# configure terminal
awplus(config)# stack 1 renumber 2
```

Validation Command **show stack**

stack renumber cascade

This command is used to renumber the members of a stack so that their IDs are ordered sequentially, relative to the member's physical position within the stack.

Caution

Changing the stack numbering will upset the existing stack member configurations such as port settings. This command is intended for use when the stack is either initially commissioned, or has undergone a major reconfiguration. In this situation you run the stack renumber command (which will automatically reboot the switch), then configure the stack members to meet the new requirements.

Syntax

```
stack <stack-ID> renumber cascade [<stack-ID>]
```

Parameter	Description
<stack-ID>	The ID of the stack member to start renumbering from, from 1 to 4.
`renumber`	Change the existing stack-ID.
`cascade`	Renumber the existing stack-ID in cascade order.
<stack-ID>	The new ID for the first member renumbered, from 1 to 4.

Default If no stack-ID is specified, the member will take the default ID of 1.

Mode Global Configuration

Usage This command is used to renumber the members of a stack so that their stack-IDs are ordered sequentially. This would normally be done either when the stack is initially configured or following a major reconfiguration.

The renumber will start on the specified stack member. If that stack-ID is not used by any of the existing stack member, the command will be rejected.

The starting stack member will be renumbered with the new stack-ID specified, or the default of member ID of 1. The stack-ID of the next physically will be the starting members ID +1, for example member ID 2. This renumbering will continue in cascading order around the stack members.

The changes will take place immediately and reboot all stack members. For this reason a confirmation prompt follows this command entry, asking whether you are sure you want to renumber and reboot the entire stack.

Example

```
awplus(config)# stack 1 renumber cascade
Any existing interface configuration may no longer be valid
Are you sure you want to renumber and reboot the entire stack?(y/n):
y
```

Related Commands

show stack
switch provision (stack)
stack renumber

stack resiliencylink

This command configures the resiliency link used by the stack. The interface used may be either an eth0 port or is a dedicated VLAN (resiliencylink VLAN) to which switch ports may become members. This VLAN is dedicated to the resiliency link function and must not be the stack management VLAN.

Syntax

```
stack resiliencylink <interface>
no stack resiliencylink
```

Parameters	Description
<interface>	The name of the interface that is connected to the resiliency link.

Mode Global Configuration

Usage The resiliency-link is only used when a backup member loses connectivity with the master via the stacking cables. Such a communication loss would occur if:

- a stacking link is removed or fails
- two or more stacking link cables are unplugged or fail
- the stack master itself fails due to a reboot or power failure

The resiliency-link allows the backup member to determine if the master is still present in the network by the reception of healthcheck messages sent by the master over the resiliency-link interface.

Reply healthcheck messages are received if the master is still online, but the stack will now split into two different “stubs”. The stub containing the existing master will continue operating as normal. The members in the masterless stub will now use a “type stack disabled-master” trigger to run a configuration to form a second temporary stack. This utilizes the remaining stack members' resources without conflicting directly with the master's configuration. If no “type stack disabled-master” trigger was configured on the switches, then the masterless stub members will disable their switch ports.

If no healthcheck messages are received, then the master is assumed to be completely offline, and so the other stack members can safely take over the master's configuration.

Caution

The purpose of the resiliency link is to enable the stack members (particularly the backup master) to check the status of the master under fault conditions. If the resiliency link is not configured, and the master loses communication with its other stack members, then the stack will assume the master is NOT present in the network, which could cause network conflicts if the master is still online. Note that this is a change to the stacking of releases prior to version 5.3.1.

Example To set the resiliency link to be VLAN 4093.

First use the **stack resiliencylink** command to create the resiliency `vlan 4093`

```
awplus# configure terminal
awplus(config)# stack resiliencylink vlan4093
```

Next use the **switchport resiliencylink** command to assign the resiliencylink vlan to the interface port, in this case `port1.0.1`.

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# switchport resiliencylink
```

Related Commands

show stack
switch provision (stack)
show stack resiliencylink
stack disabled-master-monitoring
switchport resiliencylink

stack software-auto-synchronize

This command re-enables the software version auto-synchronization feature either on a specified stack member or all stack members.

Use the **no** variant of this command to turn the software version auto synchronization feature off.

Syntax

```
stack {all|<stack-ID>} software-auto-synchronize
no stack {all|<stack-ID>} software-auto-synchronize
```

Parameter	Description
`all`	All stack members.
<stack-ID>	Stack member number, from 1 to 4.

Default All the stack members have the stack software-auto-synchronize feature enabled by default.

Mode Global Configuration

Usage This command is used to enable the software version auto-synchronization feature for either a specific stack member or all stack members and candidates.

Note that if a device attempts to join a stack but is running a software release that is different to the other stack members, the software version auto-synchronization feature will copy the master's software release onto the new member. If the software version auto-synchronization feature is not enabled, then the device will be unable to join the stack.

Note that the software version auto-synchronization feature may also result in the stack member downgrading its software release if the master is running an older software version.

Examples To turn on the software-auto-synchronize feature on stack member 2, which was previously turned off, use the following commands:

```
awplus# configure terminal
awplus(config# stack 2 software-auto-synchronize
```

To turn on the software-auto-synchronize feature for all stack members, which were previously turned off, use the following commands:

```
awplus# configure terminal
awplus(config)# stack all software-auto-synchronize
```

Validation Command **show stack**

stack virtual-chassis-id

This command specifies the stack virtual chassis ID. The ID selected will determine which virtual MAC address the stack will use. The MAC address assigned to a stack must be unique within its network.

Note The command will not take effect until the switch has been rebooted.

Syntax `stack virtual-chassis-id <id>`

Parameter	Description
<id>	The value of the ID - enter a number in the range 0 to 4095.

Mode Global Configuration

Usage The virtual-chassis-id entered will form the last 12 bits of a pre-selected MAC prefix component; that is, `0000.cd37.0xxx`. If you enable the stack virtual MAC address feature (by using the **stack virtual-mac** command) without using the **stack virtual-chassis-id** command to select the virtual-chassis-id, then the stack will select a virtual-chassis-id from a number within the assigned range.

Example To set the stack virtual-chassis-id to 63 use the commands

```
awplus# configure terminal
awplus(config)# stack virtual-chassis-id 63
```

This will result in a virtual MAC address of: `0000.cd37.003f`.

Related Commands
show running-config
show stack
switch provision (stack)
stack virtual-mac

stack virtual-mac

This command enables the stack virtual MAC address feature. For more information on this topic refer to **"Fixed or Virtual MAC Addressing" on page 110.12**. With this command set, the value applied for the virtual MAC address is determined by the setting of the command **stack virtual-chassis-id** command on page 111.31.

Caution

Stack operation is only supported if stack virtual-mac is enabled.
Before enabling the virtual MAC address feature, you should check that the stack's virtual-chassis-id is not already used by another stack in the network. Otherwise the duplicate MAC addresses will cause problems for the network traffic.

Syntax

```
stack virtual-mac
no stack virtual mac
```

Mode Global Configuration

Usage Note that this command will not take effect until the switch has been rebooted.

Example

```
awplus# configure terminal
awplus(config)# stack virtual mac
```

Related Commands

show running-config
show stack
switch provision (stack)
stack virtual-chassis-id

switch provision (stack)

This command enables you to provide the configuration for a new stack member switch prior to physically connecting it to the stack. To run this command, the stack position must be vacant. The selected hardware type must be compatible existing stack hardware.

Use the **no** variant of this command to remove an existing switch provision.

Syntax

```
switch <stack-ID> {provision|reprovision}{x510-28|x510-52}
no switch <stack-ID> provision
```

Parameter	Description
`<stack-ID>`	Stack member number, from 1 to 4.
`provision`	Provides settings within the stack configuration ready for a specific switch type to become a stack member.
`reprovision`	Reconfigure an existing provision configuration.
`x510-28`	Provision an x510-28switch.
`x510-52`	Provision an x510-52 switch.

Mode Global Configuration

Usage Note that although the syntax appears to allow provisioning on up to 8 stackable switches, in practice a maximum of 4 are configurable. Normally the stack members would be numbered 1 to 4, and so the command could be run to provision any stack member within this range; and we advise this procedure. In effect, the syntax then becomes:

```
switch <1-4> {provision|reprovision}{x510-28|x510-52}
```

However, you could number the stack units with any numbers between 1-8. For example you could number your four stack members 1, 2, 7 and 8. In this case you could provision any of the stack members within this range. We advise against numbering your stacks in this way.

Examples To provision an x510-28 switch as stack member 3, use the following commands:

```
awplus# configure terminal
awplus(config)# switch 3 provision x510-28
```

To remove the provision of the x510-28 switch as stack member 3, use the following commands:

```
awplus# configure terminal
awplus(config)# no switch 3 provision
```

Related Commands

show provisioning (stack)
show stack
switch provision (stack)

switchport resiliencylink

This command configures the switch port to be a member of the stack resiliency link VLAN. Note that this switchport will only be used for stack resiliency-link traffic and will not perform any other function, or carry any other traffic.

The **no** variant of this command removes the switchport from the resiliency link VLAN.

Syntax

```
switchport resiliencylink
no switchport resiliencylink
```

Mode Interface Configuration

Usage Note that a resiliency link cannot be part of a static or dynamic aggregator group.

Examples To set the resiliency link to be VLAN 4093:

First, use the **stack resiliencylink** command to create the resiliency `vlan4093`

```
awplus# configure terminal
awplus(config)# stack resiliencylink vlan4093
```

Next, use the **switchport resiliencylink** command to assign the resiliencylink vlan to the port, in this case `port1.0.1`.

```
awplus# configure terminal
awplus(config)# interface port1.0.1
awplus(config-if)# switchport resiliencylink
```

Related Commands

stack resiliencylink
show stack resiliencylink

vlan mode stack-local-vlan

This command enables you to create stack-local-VLANs and use ICMP to monitor and diagnose issues within specific members of the stack. When a VLAN is added using this method, all its traffic will be trapped to and processed by the CPU of the specific local stack member, rather than the CPU of the stack master.

The **no** variant of this command destroys the specified VLAN.

Syntax

```
vlan <vid> mode stack-local-vlan <member-id>
no vlan <vid>
```

Parameter	Description
<vid>	The VID of the VLAN to be created in the range 2-4094. We recommend that the first stack-local-vlan be assigned the number 4001 for the first stack member, then incremented by one for each stack member. So a stack of four members would be assigned the following VID numbers: stack member one VID 4001 stack member two VID 4002 stack member three VID 4003 stack member four VID 4004
`mode stack-local-vlan`	Specifies that the new VLAN will function as a stack-local-VLAN.
<member-id>	Specifies the new stack member ID. Enter a decimal number in the range 1-8.

Default By default, VLANs are automatically enabled as they are added.

Mode VLAN Configuration

Examples To add a stack-local-VLAN with the VID of 4002 and assign it to stack member 2, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# vlan 4002 mode stack-local-vlan 2
```

To remove VLAN 4002, use the following commands:

```
awplus# configure terminal
awplus(config)# vlan database
awplus(config-vlan)# no vlan 4002
```

undebug stack

This command applies the functionality of the **no debug stack** command on page 111.4.

Appendix A: Command List

A

aaa accounting auth-mac default 67.2
aaa accounting auth-web default 67.4
aaa accounting commands 67.6
aaa accounting dot1x 67.8
aaa accounting login 67.10
aaa accounting update 67.12
aaa authentication auth-mac 67.13
aaa authentication auth-web 67.14
aaa authentication dot1x 67.15
aaa authentication enable default group tacacs+ 67.16
aaa authentication enable default local 67.18
aaa authentication login 67.19
aaa group server 67.22
aaa local authentication attempts lockout-time 67.23
aaa local authentication attempts max-fail 67.24
abr-type 44.3
accept-lifetime 38.2
access-group 58.4
access-list extended (named) 59.4
access-list hardware (named) 58.18
access-list standard (named) 59.27
access-list (extended numbered) 59.12
access-list (hardware IP numbered) 58.6
access-list (hardware MAC numbered) 58.15
access-list (standard numbered) 59.29
accounting login 67.25
activate 13.2
active (ping-polling) 105.3
active (trigger) 103.2
address prefix 92.3
address range 92.5
advertisement-interval 82.3
aggregate-address (IPv6 RIPng) 40.3
alliedware-behavior 38.4
area authentication ipsec spi 44.4
area authentication 42.3
area default-cost (IPv6 OSPF) 44.6
area default-cost 42.5
area encryption ipsec spi esp 44.7
area filter-list 42.6
area nssa 42.7
area range (IPv6 OSPF) 44.10
area range 42.9
area stub (IPv6 OSPF) 44.11

area stub....42.11
area virtual-link authentication ipsec spi....44.14
area virtual-link encryption ipsec spi....44.16
area virtual-link (IPv6 OSPF)....44.12
area virtual-link....42.12
arp log....29.6
arp opportunistic-nd....29.9
arp security violation....80.3
arp security....80.2
arp (IP address MAC)....29.5
arp-aging-timeout....29.4
arp-mac-disparity....29.3
atmf backup bandwidth....86.5
atmf backup delete....86.6
atmf backup enable....86.7
atmf backup now....86.8
atmf backup server....86.10
atmf backup stop....86.12
atmf backup synchronize....86.12
atmf backup....86.4
atmf cleanup....86.13
atmf distribute firmware....86.14
atmf domain vlan....86.16
atmf enable....86.17
atmf group (membership)....86.18
atmf log-verbose....86.20
atmf management subnet....86.21
atmf management vlan....86.23
atmf master....86.25
atmf network-name....86.26
atmf provision node clone....86.28
atmf provision node configure boot config....86.30
atmf provision node configure boot system....86.31
atmf provision node create....86.32
atmf provision node delete....86.34
atmf provision node license-cert....86.36
atmf provision node locate....86.38
atmf provision....86.27
atmf reboot-rolling....86.39
atmf recover led-off....86.44
atmf recover....86.43
atmf remote-login....86.45
atmf restricted-login....86.46
atmf virtual-link id ip remote-id remote-ip....86.47
atmf working-set....86.49
attribute....73.2
auth auth-fail vlan....65.3
auth critical....65.5
auth dynamic-vlan-creation....65.6
auth guest-vlan....65.9
auth host-mode....65.11
auth log....65.13
auth max-supplicant....65.15
auth reauthentication....65.16
auth roaming disconnected....65.17
auth roaming enable....65.19
auth supplicant-mac....65.21

auth timeout connect-timeout 65.23
auth timeout quiet-period 65.24
auth timeout reauth-period 65.25
auth timeout server-timeout 65.26
auth timeout supp-timeout 65.27
auth two-step enable 65.28
authentication 73.5
auth-mac enable 65.31
auth-mac method 65.32
auth-mac password 65.33
auth-mac reauth-relearning 65.34
auth-web enable 65.35
auth-web forward 65.36
auth-web max-auth-fail 65.38
auth-web method 65.39
auth-web-server blocking-mode 65.40
auth-web-server dhcp ipaddress 65.41
auth-web-server dhcp lease 65.42
auth-web-server dhcp-wpad-option 65.43
auth-web-server gateway (deleted) 65.44
auth-web-server host-name 65.45
auth-web-server http-redirect (deleted) 65.45
auth-web-server intercept-port 65.46
auth-web-server ipaddress 65.47
auth-web-server login-url 65.48
auth-web-server mode (deleted) 65.49
auth-web-server page logo 65.49
auth-web-server page sub-title 65.50
auth-web-server page success-message 65.51
auth-web-server page title 65.52
auth-web-server page welcome-message 65.53
auth-web-server ping-poll enable 65.54
auth-web-server ping-poll failcount 65.55
auth-web-server ping-poll interval 65.56
auth-web-server ping-poll reauth-timer-refresh 65.57
auth-web-server ping-poll timeout 65.58
auth-web-server port 65.59
auth-web-server redirect-delay-time 65.60
auth-web-server redirect-url 65.61
auth-web-server session-keep 65.62
auth-web-server ssl intercept-port 65.64
auth-web-server sslport (deleted) 65.63
auth-web-server ssl 65.63
autoboot enable 7.5
auto-cost reference bandwidth (IPv6 OSPF) 44.19
auto-cost reference bandwidth 42.14

B

backpressure 17.2
bandwidth (duplicate) 44.21
bandwidth 42.16
banner exec 10.2
banner login (SSH) 78.2
banner login (system) 10.4
banner motd 10.5

boot config-file backup 7.8
boot config-file 7.6
boot system backup 7.11
boot system 7.9
bootfile 90.2

C

capability opaque 42.17
capability restart 42.17
cd 7.12
channel-group 23.3
circuit-failover 82.5
cisco-metric-behavior (RIP) 38.6
class-map 63.3
class 63.2
clear aaa local user lockout 67.26
clear arp security statistics 80.4
clear arp-cache 29.10
clear atmf links statistics 86.51
clear counter ipv6 dhcp-client 92.7
clear counter ipv6 dhcp-server 92.7
clear counter stack 111.3
clear exception log 12.2
clear gvrp statistics 27.2
clear ip dhcp binding 90.3
clear ip dhcp snooping binding 80.5
clear ip dhcp snooping statistics 80.6
clear ip dns forwarding cache 29.10
clear ip igmp group 49.3
clear ip igmp interface 49.4
clear ip igmp 49.2
clear ip mroute pim sparse-mode 51.3
clear ip mroute statistics 47.7
clear ip mroute 47.6
clear ip ospf process 42.18
clear ip pim sparse-mode bsr rp-set * 51.2
clear ip prefix-list 59.35
clear ip rip route 38.7
clear ipv6 dhcp binding 92.8
clear ipv6 dhcp client 92.9
clear ipv6 mld group 56.8
clear ipv6 mld interface 56.8
clear ipv6 mld 56.7
clear ipv6 mroute pim sparse-mode 53.3
clear ipv6 mroute pim 53.3
clear ipv6 mroute statistics 47.9
clear ipv6 mroute 47.8
clear ipv6 neighbors 31.2
clear ipv6 ospf process 44.22
clear ipv6 pim sparse-mode bsr rp-set * 53.4
clear ipv6 rip route 40.4
clear lacp counters 23.5
clear line console 5.2
clear line vty 5.3
clear lldp statistics 97.2

clear lldp table....97.3
clear log buffered....12.3
clear log permanent....12.3
clear log....12.2
clear loop-protection counters....17.3
clear mac address-table dynamic....17.5
clear mac address-table static....17.4
clear mls qos interface policer-counters....63.4
clear ping-poll....105.4
clear port counter....17.7
clear power-inline counters interface....25.3
clear radius local-server statistics....73.6
clear spanning-tree detected protocols (RSTP and MSTP)....21.4
clear spanning-tree statistics....21.3
clear ssh....78.3
clear test cable-diagnostics tdr....109.2
clear test interface....15.2
clear vlan statistics....19.2
clock set....10.6
clock summer-time date....10.7
clock summer-time recurring....10.8
clock timezone....10.10
commit (IPv4)....58.34
commit (IPv6)....60.4
compatible rfc1583....42.18
configure terminal....4.2
continuous-reboot-prevention....10.11
copy current-software....7.12
copy debug....7.13
copy fdb-radius-users (to file)....73.7
copy local-radius-user-db (from file)....73.9
copy local-radius-user-db (to file)....73.10
copy proxy-autoconfig-file....65.65
copy running-config....7.14
copy startup-config....7.15
copy web-auth-https-file....65.65
copy zmodem....7.18
copy (URL)....7.16
create autoboot....7.19
critical-interval....105.5
crypto key destroy hostkey....78.4
crypto key destroy userkey....78.5
crypto key generate hostkey....78.6
crypto key generate userkey....78.7
crypto key pubkey-chain knownhosts....78.8
crypto key pubkey-chain userkey....78.10
crypto pki enroll local local-radius-all-users....73.12
crypto pki enroll local user....73.13
crypto pki enroll local....73.11
crypto pki export local pem....73.14
crypto pki export local pkcs12....73.15
crypto pki trustpoint local....73.16

D

day 103.3
deadtime (RADIUS server group) 71.2
debug aaa 67.27
debug arp security 80.6
debug atmf packet 86.54
debug atmf 86.52
debug crypto pki 73.17
debug dot1x 69.2
debug epsr 84.3
debug gvrp 27.3
debug igmp 49.5
debug ip dhcp snooping 80.7
debug ip dns forwarding 29.11
debug ip packet interface 29.11
debug ipv6 ospf events 44.22
debug ipv6 ospf ifsm 44.23
debug ipv6 ospf lsa 44.24
debug ipv6 ospf nfsm 44.25
debug ipv6 ospf packet 44.26
debug ipv6 ospf route 44.27
debug ipv6 pim sparse-mode packet 53.7
debug ipv6 pim sparse-mode timer 53.8
debug ipv6 pim sparse-mode 53.5
debug ipv6 rip 40.5
debug lacp 23.6
debug lldp 97.4
debug loopprot 17.8
debug mail 98.2
debug mld 56.9
debug mstp (RSTP and STP) 21.5
debug nsm mcast6 ..... 47.11
debug nsm mcast 47.10
debug ospf events 42.19
debug ospf ifsm 42.20
debug ospf lsa 42.21
debug ospf nfsm 42.22
debug ospf nsm 42.23
debug ospf packet 42.24
debug ospf route 42.25
debug pim dense-mode all 55.2
debug pim dense-mode context 55.3
debug pim dense-mode decode 55.4
debug pim dense-mode encode 55.4
debug pim dense-mode fsm 55.5
debug pim dense-mode mrt 55.6
debug pim dense-mode nexthop 55.7
debug pim dense-mode nsm 55.8
debug pim dense-mode vif 55.9
debug pim sparse-mode timer 51.5
debug pim sparse-mode 51.4
debug ping-poll 105.6
debug platform packet 17.9
debug power-inline 25.4
debug radius 71.3

debug rip....38.8
debug sflow agent....107.3
debug sflow....107.2
debug snmp....94.2
debug ssh client....78.12
debug ssh server....78.13
debug stack....111.4
debug trigger....103.4
debug vrrp events....82.8
debug vrrp packet....82.9
debug vrrp....82.7
default log buffered....12.4
default log console....12.4
default log email....12.5
default log host....12.5
default log monitor....12.6
default log permanent....12.6
default-action....63.5
default-information originate (IPv6 RIPng)....40.6
default-information originate (OSPF)....42.26
default-information originate (RIP)....38.9
default-metric (IPv6 OSPF)....44.28
default-metric (IPv6 RIPng)....40.7
default-metric (OSPF)....42.27
default-metric (RIP)....38.10
default-router....90.4
delete debug....7.21
delete mail....98.3
delete....7.20
description (interface)....14.2
description (ping-polling)....105.7
description (QoS policy-map)....63.6
description (trigger)....103.5
dir....7.22
disable (Privileged Exec mode)....4.2
disable (VRRP)....82.10
distance (IPv6 OSPF)....44.29
distance (OSPF)....42.28
distance (RIP)....38.11
distribute-list (IPv6 OSPF)....44.31
distribute-list (IPv6 RIPng)....40.8
distribute-list (OSPF)....42.30
distribute-list (RIP)....38.12
dns-server (DHCPv6)....92.10
dns-server....90.5
domain-name (DHCPv6)....92.11
domain-name....90.6
domain-style....73.18
dos....59.36
dot1x control-direction....69.3
dot1x eapol-version....69.5
dot1x eap....69.4
dot1x initialize interface....69.6
dot1x initialize supplicant....69.7
dot1x keytransmit....69.8
dot1x max-auth-fail....69.9
dot1x max-reauth-req....69.10

dot1x port-control 69.11
dot1x timeout tx-period 69.13
do 4.3
duplex 17.11

E

echo 13.3
ecofriendly led 10.13
ecofriendly lpi 10.14
edit (URL) 7.25
edit 7.24
egress-rate-limit 63.7
egress-vlan-id 73.19
egress-vlan-name 73.20
enable db-summary-opt 42.32
enable password 5.4
enable secret 5.7
enable (Privileged Exec mode) 4.4
enable (VRRP) 82.11
end 4.6
epsr configuration 84.5
epsr datavlan 84.6
epsr enhancedrecovery enable 84.7
epsr mode master controlvlan primary port 84.8
epsr mode transit controlvlan 84.9
epsr priority 84.10
epsr state 84.11
epsr trap 84.12
epsr 84.4
erase factory-default 86.57
erase proxy-autoconfig-file 65.66
erase startup-config 7.26
erase web-auth-https-file 65.66
exception coredump size (deleted) 12.7
exec-timeout 5.10
exit 4.6

F

fail-count 105.8
findme 10.16
flowcontrol hardware (asyn/console) 5.11
flowcontrol (switch port) 17.12
fullupdate (RIP) 38.13

G

group 73.21
gvrp dynamic-vlan-creation 27.5
gvrp enable (global) 27.6
gvrp registration 27.7
gvrp timer 27.8
gvrp (interface) 27.4

H

help 4.7
host area 42.33
hostname 10.18
host 90.7

I

instance priority (MSTP) 21.8
instance vlan (MSTP) 21.9
interface tunnel 33.2
interface (to configure) 14.3
ip address dhcp 90.8
ip address 29.13
ip dhcp bootp ignore 90.9
ip dhcp leasequery enable 90.10
ip dhcp option 90.11
ip dhcp pool 90.13
ip dhcp snooping agent-option allow-untrusted 80.10
ip dhcp snooping agent-option circuit-id vlantriplet 80.11
ip dhcp snooping agent-option remote-id 80.12
ip dhcp snooping agent-option 80.9
ip dhcp snooping binding 80.13
ip dhcp snooping database 80.14
ip dhcp snooping delete-by-client 80.15
ip dhcp snooping delete-by-linkdown 80.16
ip dhcp snooping max-bindings 80.17
ip dhcp snooping subscriber-id 80.18
ip dhcp snooping trust 80.19
ip dhcp snooping verify mac-address 80.20
ip dhcp snooping violation 80.21
ip dhcp snooping 80.8
ip dhcp-relay agent-option checking 90.15
ip dhcp-relay agent-option remote-id 90.16
ip dhcp-relay agent-option 90.14
ip dhcp-relay information policy 90.17
ip dhcp-relay maxhops 90.19
ip dhcp-relay max-message-length 90.20
ip dhcp-relay server-address 90.22
ip directed-broadcast 29.15
ip dns forwarding cache 29.17
ip dns forwarding dead-time 29.18
ip dns forwarding retry 29.19
ip dns forwarding source-interface 29.20
ip dns forwarding timeout 29.21
ip dns forwarding 29.16
ip domain-list 29.22
ip domain-lookup 29.23
ip domain-name 29.24
ip forward-protocol udp 29.25
ip gratuitous-arp-link 29.27
ip helper-address 29.28
ip igmp access-group 49.7
ip igmp immediate-leave 49.8
ip igmp last-member-query-count 49.9

ip igmp last-member-query-interval 49.10
ip igmp limit 49.11
ip igmp mroute-proxy 49.13
ip igmp proxy-service 49.14
ip igmp querier-timeout 49.15
ip igmp query-holdtime 49.16
ip igmp query-interval 49.18
ip igmp query-max-response-time 49.20
ip igmp ra-option (Router Alert) 49.22
ip igmp robustness-variable 49.23
ip igmp snooping fast-leave 49.25
ip igmp snooping mrouter 49.26
ip igmp snooping querier 49.27
ip igmp snooping report-suppression 49.28
ip igmp snooping routermode 49.29
ip igmp snooping tcn query solicit 49.31
ip igmp snooping 49.24
ip igmp source-address-check 49.33
ip igmp ssm-map enable 49.35
ip igmp ssm-map static 49.36
ip igmp ssm 49.34
ip igmp startup-query-count 49.40
ip igmp startup-query-interval 49.41
ip igmp static-group 49.38
ip igmp version 49.42
ip igmp 49.6
ip local-proxy-arp 29.30
ip mroute 47.12
ip multicast forward-first-packet 47.14
ip multicast route-limit 47.17
ip multicast route 47.15
ip multicast wrong-vif-suppression 47.18
ip multicast-routing 47.19
ip name-server 29.31
ip ospf authentication-key 42.35
ip ospf authentication 42.34
ip ospf cost 42.36
ip ospf database-filter 42.37
ip ospf dead-interval 42.38
ip ospf disable all 42.39
ip ospf hello-interval 42.40
ip ospf message-digest-key 42.41
ip ospf mtu-ignore 42.43
ip ospf mtu 42.42
ip ospf network 42.44
ip ospf priority 42.45
ip ospf resync-timeout 42.46
ip ospf retransmit-interval 42.47
ip ospf transmit-delay 42.48
ip pim accept-register list 51.7
ip pim anycast-rp 51.8
ip pim bsr-border 51.9
ip pim bsr-candidate 51.10
ip pim cisco-register-checksum group-list 51.12
ip pim cisco-register-checksum 51.11
ip pim crp-cisco-prefix 51.13
ip pim dense-mode passive 55.10

ip pim dense-mode 55.10
ip pim dr-priority 51.14
ip pim exclude-genid 51.15
ip pim ext-srcs-directly-connected (PIM-DM) 55.11
ip pim ext-srcs-directly-connected (PIM-SM) 51.16
ip pim hello-holdtime (PIM-DM) 55.11
ip pim hello-holdtime (PIM-SM) 51.17
ip pim hello-interval (PIM-DM) 55.12
ip pim hello-interval (PIM-SM) 51.18
ip pim ignore-rp-set-priority 51.19
ip pim jp-timer 51.20
ip pim max-graft-retries 55.13
ip pim neighbor-filter (PIM-DM) 55.15
ip pim neighbor-filter (PIM-SM) 51.21
ip pim propagation-delay 55.16
ip pim register-rate-limit 51.22
ip pim register-rp-reachability 51.22
ip pim register-source 51.23
ip pim register-suppression 51.24
ip pim rp-address 51.25
ip pim rp-candidate 51.27
ip pim rp-register-kat 51.28
ip pim sparse-mode passive 51.30
ip pim sparse-mode 51.29
ip pim spt-threshold group-list 51.32
ip pim spt-threshold 51.31
ip pim ssm 51.33
ip pim state-refresh origination-interval 55.17
ip prefix-list 59.40
ip proxy-arp 29.32
ip radius source-interface 71.4
ip redirects 29.33
ip rip authentication key-chain 38.14
ip rip authentication mode 38.17
ip rip authentication string 38.21
ip rip receive version 38.24
ip rip receive-packet 38.23
ip rip send version 1-compatible 38.28
ip rip send version 38.26
ip rip send-packet 38.25
ip rip split-horizon 38.29
ip route 36.3
ip source binding 80.22
ip (ping-polling) 105.9
ipv6 access-list extended proto 61.8
ipv6 access-list extended (named) 61.4
ipv6 access-list standard (named) 61.16
ipv6 access-list (named) 60.5
ipv6 address autoconfig 31.5
ipv6 address dhcp 92.14
ipv6 address (DHCPv6 PD) 92.12
ipv6 address 31.3
ipv6 dhcp client pd 92.15
ipv6 dhcp option 92.16
ipv6 dhcp pool 92.18
ipv6 dhcp server 92.19
ipv6 enable 31.7

ipv6 forwarding 31.8
ipv6 local pool 92.20
ipv6 mld access-group 56.12
ipv6 mld immediate-leave 56.13
ipv6 mld last-member-query-count 56.14
ipv6 mld last-member-query-interval 56.15
ipv6 mld limit 56.16
ipv6 mld querier-timeout 56.18
ipv6 mld query-interval 56.19
ipv6 mld query-max-response-time 56.20
ipv6 mld robustness-variable 56.21
ipv6 mld snooping fast-leave 56.24
ipv6 mld snooping mrouter 56.25
ipv6 mld snooping querier 56.27
ipv6 mld snooping report-suppression 56.28
ipv6 mld snooping 56.22
ipv6 mld static-group 56.30
ipv6 mld version 56.33
ipv6 mld 56.11
ipv6 multicast route-limit 47.23
ipv6 multicast route 47.20
ipv6 multicast-routing 47.24
ipv6 nd managed-config-flag 31.9
ipv6 nd minimum-ra-interval 31.10
ipv6 nd other-config-flag 31.11
ipv6 nd prefix (DHCPv6) 92.22
ipv6 nd prefix 31.12
ipv6 nd raguard 31.16
ipv6 nd ra-interval 31.14
ipv6 nd ra-lifetime 31.15
ipv6 nd reachable-time 31.17
ipv6 nd retransmission-time 31.18
ipv6 nd suppress-ra 31.19
ipv6 neighbor 31.20
ipv6 opportunistic-nd 31.21
ipv6 ospf authentication spi 44.33
ipv6 ospf cost 44.35
ipv6 ospf dead-interval 44.36
ipv6 ospf display route single-line 44.37
ipv6 ospf encryption spi esp 44.38
ipv6 ospf hello-interval 44.41
ipv6 ospf network 44.42
ipv6 ospf priority 44.43
ipv6 ospf retransmit-interval 44.44
ipv6 ospf transmit-delay 44.45
ipv6 pim accept-register 53.10
ipv6 pim anycast-rp 53.11
ipv6 pim bsr-border 53.12
ipv6 pim bsr-candidate 53.13
ipv6 pim cisco-register-checksum group-list 53.15
ipv6 pim cisco-register-checksum 53.14
ipv6 pim crp-cisco-prefix 53.16
ipv6 pim dr-priority 53.17
ipv6 pim exclude-genid 53.18
ipv6 pim ext-srcs-directly-connected 53.19
ipv6 pim hello-holdtime 53.20
ipv6 pim hello-interval 53.21

ipv6 pim ignore-rp-set-priority 53.22
ipv6 pim jp-timer 53.23
ipv6 pim neighbor-filter 53.24
ipv6 pim register-rate-limit 53.25
ipv6 pim register-rp-reachability 53.26
ipv6 pim register-source 53.27
ipv6 pim register-suppression 53.28
ipv6 pim rp embedded 53.32
ipv6 pim rp-address 53.29
ipv6 pim rp-candidate 53.31
ipv6 pim rp-register-kat 53.33
ipv6 pim sparse-mode passive 53.35
ipv6 pim sparse-mode 53.34
ipv6 pim spt-threshold group-list 53.37
ipv6 pim spt-threshold 53.36
ipv6 pim unicast-bsm 53.38
ipv6 prefix-list 61.20
ipv6 rip metric-offset 40.9
ipv6 rip split-horizon 40.11
ipv6 router ospf area 44.46
ipv6 router rip 40.12
ipv6 route 31.22
ipv6 traffic-filter 60.17
key chain 38.31
key-string 38.32
key 38.30

L

lacp port-priority 23.7
lacp system-priority 23.8
lacp timeout 23.9
lease 90.24
length (asyn) 5.13
length (ping-poll data) 105.10
license member (deleted) 9.3
license 9.2
line 5.14
link-address 92.24
linkflap action 17.14
lldp faststart-count 97.5
lldp holdtime-multiplier 97.6
lldp management-address 97.7
lldp med-notifications 97.8
lldp med-tlv-select 97.9
lldp non-strict-med-tlv-order-check 97.11
lldp notification-interval 97.12
lldp notifications 97.13
lldp port-number-type 97.14
lldp reinit 97.15
lldp run 97.16
lldp timer 97.17
lldp tlv-select 97.18
lldp transmit receive 97.20
lldp tx-delay 97.21
location civic-location configuration 97.22

location civic-location identifier 97.26
location civic-location-id 97.27
location coord-location configuration 97.28
location coord-location identifier 97.30
location coord-location-id 97.31
location elin-location-id 97.33
location elin-location 97.32
log buffered size 12.12
log buffered (filter) 12.9
log buffered 12.8
log console (filter) 12.14
log console 12.13
log email time 12.21
log email (filter) 12.18
log email 12.17
log host time 12.27
log host (filter) 12.24
log host 12.23
log monitor (filter) 12.29
log permanent size 12.36
log permanent (filter) 12.33
log permanent 12.32
login authentication 67.28
logout 4.7
log-rate-limit nsm 12.37
loop-protection action 17.16
loop-protection timeout 17.17
loop-protection 17.15

M

mac address-table acquire 17.18
mac address-table ageing-time 17.19
mac address-table static 17.20
mac address-table thrash-limit 17.21
mail from 98.5
mail smtpserver 98.5
mail 98.4
match access-group 63.8
match cos 63.9
match dscp 63.10
match eth-format protocol 63.11
match inner-cos 63.14
match inner-vlan 63.15
match interface 46.3
match ip address 46.4
match ip next-hop 46.6
match ip-precedence 63.16
match ipv6 address 46.8
match mac-type 63.17
match metric 46.9
match route-type 46.10
match tag 46.11
match tcp-flags 63.18
match vlan 63.19
max-concurrent-dd (IPv6 OSPF) 44.47

max-concurrent-dd 42.49
max-fib-routes 10.20
maximum-access-list 59.42
maximum-area 42.50
maximum-paths 36.5
maximum-prefix 38.33
max-static-routes 10.21
mirror interface 17.22
mkdir 7.27
mls qos cos 63.20
mls qos enable 63.21
mls qos map cos-queue to 63.22
mls qos map premark-dscp to 63.23
move debug 7.29
move 7.28
mru 14.5
mtu 14.7
multicast 47.25

N

nas 73.22
neighbor (IPv6 RIPng) 40.13
neighbor (OSPF) 42.51
neighbor (RIP) 38.34
network area 42.52
network (DHCP) 90.26
network (RIP) 38.35
next-server 90.27
no debug all 10.22
no police 63.25
normal-interval 105.11
ntp access-group 88.2
ntp authenticate 88.3
ntp authentication-key 88.4
ntp broadcastdelay 88.5
ntp master 88.6
ntp peer 88.7
ntp server 88.9
ntp source 88.11
ntp trusted-key 88.13

O

offset-list (IPv6 RIPng) 40.14
offset-list (RIP) 38.36
optimistic-nd 29.34
option (DHCPv6) 92.26
option 90.28
ospf abr-type 42.53
ospf restart grace-period 42.54
ospf restart helper 42.55
ospf router-id 42.57
overflow database external 42.59
overflow database 42.58

P

passive-interface (IPv6 OSPF) 44.48
passive-interface (IPv6 RIPng) 40.15
passive-interface (OSPF) 42.60
passive-interface (RIP) 38.37
ping ipv6 ..... 31.23
ping-poll 105.12
ping 29.35
platform hwfilter-size 17.24
platform l3-vlan-hashing-algorithm 65.67
platform load-balancing 17.25
platform mac-vlan-hashing-algorithm 65.68
platform stop-unreg-mc-flooding 17.26
platform vlan-stacking-tpid 17.28
polarity 17.29
police single-rate action 63.26
police twin-rate action 63.28
policy-map 63.30
port-vlan-forwarding-priority 19.3
power-inline allow-legacy 25.5
power-inline description 25.6
power-inline enable 25.7
power-inline max 25.8
power-inline priority 25.10
power-inline usage-threshold 25.12
preempt-mode 82.12
prefix-delegation pool 92.29
priority-queue 63.31
priority 82.14
private-vlan association 19.7
private-vlan 19.6
privilege level 5.16
probe enable 90.30
probe packets 90.31
probe timeout 90.32
probe type 90.33
pwd 7.30

R

radius-server deadtime 71.5
radius-server host 71.6
radius-server key 71.10
radius-server local 73.23
radius-server retransmit 71.11
radius-server timeout 71.12
range 90.34
reboot rolling 111.5
reboot 10.23
recv-buffer-size (IPv6 RIPng) 40.16
recv-buffer-size (RIP) 38.38
redistribute (IPv6 OSPF) 44.49
redistribute (IPv6 RIPng) 40.17
redistribute (OSPF) 42.61
redistribute (RIP) 38.39

region (MSTP) 21.10
reload rolling 111.5
reload 10.23
remark new-cos 63.34
remark-map 63.32
remote-command (deleted) 111.6
remote-login 111.7
repeat 103.6
restart ipv6 ospf graceful 44.50
restart ospf graceful 42.63
restart rip graceful 38.40
revision (MSTP) 21.11
rip restart grace-period 38.41
rmdir 7.31
rmon alarm 100.2
rmon collection history 100.4
rmon collection stats 100.5
rmon event 100.6
route (IPv6 RIPng) 40.18
route (RIP) 38.42
route-map 46.12
router ipv6 ospf 44.51
router ipv6 rip 40.18
router ipv6 vrrp (interface) 82.17
router ospf 42.64
router rip 38.43
router vrrp (interface) 82.16
router-id (IPv6 OSPF) 44.52
router-id 42.65
route 90.35

S

sample-size 105.13
script 103.7
security-password forced-change 5.18
security-password history 5.17
security-password lifetime 5.19
security-password minimum-categories 5.20
security-password minimum-length 5.21
security-password reject-expired-pwd 5.22
security-password warning 5.23
send-lifetime 38.44
server auth-port 73.24
server enable 73.25
server (Server Group) 71.14
service advanced-vty 5.24
service dhcp-relay 90.36
service dhcp-server 90.37
service dhcp-snooping 80.23
service http 5.25
service password-encryption 5.26
service power-inline 25.13
service ssh 78.14
service telnet 5.27
service terminal-length 5.28

service test 15.3
service-policy input 63.36
set ip next-hop (PBR) 63.37
set ip next-hop (route map) 46.14
set metric-type 46.17
set metric 46.15
set tag 46.18
sflow agent (address) 107.4
sflow collector max-datagram-size 107.8
sflow collector (address) 107.6
sflow enable 107.9
sflow max-header-size 107.10
sflow polling-interval 107.12
sflow sampling-rate 107.13
show access-list (IPv4 Hardware ACLs) 58.35
show access-list (IPv4 Software ACLs) 59.43
show arp security interface 80.26
show arp security statistics 80.27
show arp security 80.25
show arp 29.36
show atmf backup 86.62
show atmf detail 86.64
show atmf group members 86.68
show atmf group 86.66
show atmf links detail 86.70
show atmf links statistics 86.77
show atmf links 86.69
show atmf memory 86.80
show atmf nodes 86.82
show atmf provision nodes 86.83
show atmf tech 86.84
show atmf working-set 86.86
show atmf 86.58
show auth two-step supplicant brief 65.69
show auth-mac diagnostics 65.71
show auth-mac interface 65.72
show auth-mac sessionstatistics 65.75
show auth-mac statistics interface 65.76
show auth-mac supplicant interface 65.79
show auth-mac supplicant 65.77
show auth-mac 65.70
show auth-web diagnostics 65.82
show auth-web interface 65.83
show auth-web sessionstatistics 65.86
show auth-web statistics interface 65.87
show auth-web supplicant interface 65.89
show auth-web supplicant 65.88
show auth-web-server page 65.91
show auth-web-server 65.90
show auth-web 65.80
show autoboot 7.32
show banner login 78.16
show boot 7.33
show class-map 63.38
show clock 10.24
show continuous-reboot-prevention 10.25
show counter dhcp-client 90.38

show counter dhcp-relay 90.39
show counter dhcp-server 90.43
show counter ipv6 dhcp-client 92.31
show counter ipv6 dhcp-server 92.32
show counter log 12.38
show counter mail 98.6
show counter ntp 88.14
show counter ping-poll 105.14
show counter snmp-server 94.4
show counter stack 111.8
show cpu history 10.29
show cpu 10.26
show crypto key hostkey 78.17
show crypto key pubkey-chain knownhosts 78.18
show crypto key pubkey-chain userkey 78.19
show crypto key userkey 78.20
show crypto pki certificates local-radius-all-users 73.28
show crypto pki certificates user 73.29
show crypto pki certificates 73.26
show crypto pki trustpoints 73.30
show debugging aaa 67.29
show debugging arp security 80.29
show debugging atmf packet 86.87
show debugging atmf 86.86
show debugging dot1x 69.14
show debugging epsr 84.12
show debugging gvrp 27.10
show debugging igmp 49.43
show debugging ip dhcp snooping 80.30
show debugging ip dns forwarding 29.37
show debugging ip packet 29.38
show debugging ipv6 ospf 44.53
show debugging ipv6 pim sparse-mode 53.39
show debugging ipv6 rip 40.19
show debugging lacp 23.11
show debugging lldp 97.34
show debugging loopprot 17.30
show debugging mld 56.34
show debugging mstp 21.12
show debugging ospf 42.66
show debugging pim dense-mode 55.18
show debugging pim sparse-mode 51.34
show debugging platform packet 17.30
show debugging power-inline 25.14
show debugging radius 71.16
show debugging rip 38.45
show debugging sflow 107.14
show debugging snmp 94.7
show debugging stack 111.12
show debugging trigger 103.9
show debugging vrrp 82.18
show debugging 10.31
show dhcp lease 90.45
show diagnostic channel-group 23.12
show dos interface 59.45
show dot1x diagnostics 69.17
show dot1x interface 69.18

show dot1x sessionstatistics 69.23
show dot1x statistics interface 69.24
show dot1x supplicant interface 69.27
show dot1x supplicant 69.25
show dot1x 69.15
show ecofriendly 10.32
show epsr common segments 84.18
show epsr config-check 84.19
show epsr counters 84.21
show epsr summary 84.22
show epsr <epsr-instance> 84.20
show epsr 84.13
show etherchannel detail 23.14
show etherchannel summary 23.15
show etherchannel 23.13
show exception log 12.39
show file systems 7.36
show file 7.35
show flowcontrol interface 17.31
show gvrp configuration 27.11
show gvrp machine 27.12
show gvrp statistics 27.13
show gvrp timer 27.14
show history 4.8
show hosts 29.39
show interface access-group 58.37
show interface brief 14.12
show interface memory 10.34
show interface status 14.13
show interface switchport 17.32
show interface 14.9
show ip access-list 59.47
show ip dhcp binding 90.46
show ip dhcp pool 90.47
show ip dhcp server statistics 90.51
show ip dhcp server summary 90.53
show ip dhcp snooping acl 80.32
show ip dhcp snooping agent-option 80.34
show ip dhcp snooping binding 80.36
show ip dhcp snooping interface 80.37
show ip dhcp snooping statistics 80.39
show ip dhcp snooping 80.31
show ip dhcp-relay 90.50
show ip dns forwarding cache 29.41
show ip dns forwarding server 29.42
show ip dns forwarding 29.40
show ip domain-list 29.43
show ip domain-name 29.43
show ip igmp groups 49.44
show ip igmp interface 49.45
show ip igmp proxy 49.48
show ip igmp snooping mrouter 49.49
show ip igmp snooping routermode 49.50
show ip igmp snooping statistics 49.51
show ip interface 29.44
show ip mroute 47.26
show ip mvif 47.28

show ip name-server 29.45
show ip ospf border-routers 42.69
show ip ospf database asbr-summary 42.72
show ip ospf database external 42.73
show ip ospf database network 42.75
show ip ospf database nssa-external 42.77
show ip ospf database opaque-area 42.79
show ip ospf database opaque-as 42.80
show ip ospf database opaque-link 42.81
show ip ospf database router 42.82
show ip ospf database summary 42.83
show ip ospf database 42.70
show ip ospf interface 42.85
show ip ospf neighbor 42.86
show ip ospf route 42.88
show ip ospf virtual-links 42.89
show ip ospf 42.67
show ip pim dense-mode interface detail 55.20
show ip pim dense-mode interface 55.19
show ip pim dense-mode mroute 55.21
show ip pim dense-mode neighbor detail 55.23
show ip pim dense-mode neighbor 55.22
show ip pim dense-mode nexthop 55.24
show ip pim sparse-mode bsr-router 51.34
show ip pim sparse-mode interface detail 51.36
show ip pim sparse-mode interface 51.35
show ip pim sparse-mode local-members 51.37
show ip pim sparse-mode mroute detail 51.40
show ip pim sparse-mode mroute 51.38
show ip pim sparse-mode neighbor 51.42
show ip pim sparse-mode nexthop 51.43
show ip pim sparse-mode rp mapping 51.44
show ip pim sparse-mode rp-hash 51.44
show ip prefix-list 59.48
show ip protocols ospf 42.90
show ip protocols rip 38.45
show ip rip database 38.47
show ip rip interface 38.47
show ip rip 38.46
show ip route database 36.8
show ip route summary 36.10
show ip route 36.6
show ip rpf 47.28
show ip sockets 29.46
show ip source binding 80.42
show ip traffic 29.48
show ipv6 access-list (IPv6 Hardware ACLs) 60.18
show ipv6 access-list (IPv6 Software ACLs) 61.22
show ipv6 dhcp binding 92.34
show ipv6 dhcp interface 92.36
show ipv6 dhcp pool 92.38
show ipv6 dhcp 92.33
show ipv6 forwarding 31.24
show ipv6 interface brief 31.25
show ipv6 mif 47.31
show ipv6 mld groups 56.35
show ipv6 mld interface 56.36

show ipv6 mld snooping mrouter 56.37
show ipv6 mld snooping statistics 56.38
show ipv6 mroute 47.29
show ipv6 neighbors 31.25
show ipv6 ospf database external 44.57
show ipv6 ospf database grace 44.59
show ipv6 ospf database inter-prefix 44.61
show ipv6 ospf database inter-router 44.63
show ipv6 ospf database intra-prefix 44.65
show ipv6 ospf database link 44.67
show ipv6 ospf database network 44.69
show ipv6 ospf database router 44.70
show ipv6 ospf database 44.55
show ipv6 ospf interface 44.74
show ipv6 ospf neighbor 44.76
show ipv6 ospf route 44.77
show ipv6 ospf virtual-links 44.78
show ipv6 ospf 44.54
show ipv6 pim sparse-mode bsr-router 53.40
show ipv6 pim sparse-mode interface detail 53.42
show ipv6 pim sparse-mode interface 53.41
show ipv6 pim sparse-mode local-members 53.43
show ipv6 pim sparse-mode mroute detail 53.46
show ipv6 pim sparse-mode mroute 53.44
show ipv6 pim sparse-mode neighbor 53.48
show ipv6 pim sparse-mode nexthop 53.49
show ipv6 pim sparse-mode rp mapping 53.51
show ipv6 pim sparse-mode rp nexthop 53.52
show ipv6 pim sparse-mode rp-hash 53.50
show ipv6 prefix-list 61.23
show ipv6 protocols rip 40.19
show ipv6 rip database 40.21
show ipv6 rip interface 40.22
show ipv6 rip 40.20
show ipv6 route summary 31.28
show ipv6 route 31.26
show lacp sys-id 23.16
show lacp-counter 23.17
show license brief member 9.10
show license brief 9.6
show license member 9.8
show license 9.4
show lldp interface 97.37
show lldp local-info 97.39
show lldp neighbors detail 97.45
show lldp neighbors 97.43
show lldp statistics interface 97.49
show lldp statistics 97.48
show lldp 97.35
show location 97.51
show log config 12.42
show log permanent 12.45
show log 12.40
show loop-protection 17.33
show mac address-table thrash-limit 17.36
show mac address-table 17.34
show mail 98.6

show memory allocations 10.38
show memory history 10.39
show memory pools 10.41
show memory shared 10.42
show memory 10.36
show mirror interface 17.38
show mirror 17.37
show mls qos interface policer-counters 63.42
show mls qos interface queue-counters 63.43
show mls qos interface storm-status 63.44
show mls qos interface 63.40
show mls qos maps cos-queue 63.45
show mls qos maps premark-dscp 63.46
show mls qos 63.39
show ntp associations 88.15
show ntp status 88.16
show ping-poll 105.16
show platform classifier statistics utilization brief 17.40
show platform port 17.41
show platform 17.39
show policy-map 63.47
show port etherchannel 23.18
show port-security interface 17.47
show port-security intrusion 17.48
show port-vlan-forwarding-priority 19.8
show power-inline counters 25.18
show power-inline interface detail 25.22
show power-inline interface 25.20
show power-inline 25.15
show privilege 5.29
show process 10.43
show provisioning (stack) 111.14
show proxy-autoconfig-file 65.92
show radius local-server group 73.31
show radius local-server nas 73.32
show radius local-server statistics 73.33
show radius local-server user 73.34
show radius statistics 71.19
show radius 71.17
show reboot history 10.45
show rmon alarm 100.7
show rmon event 100.8
show rmon history 100.9
show rmon statistics 100.10
show route-map 46.19
show router-id 10.46
show running-config access-list 7.39
show running-config as-path access-list 7.40
show running-config atmf 86.87
show running-config dhcp 7.41
show running-config full 7.42
show running-config interface 7.43
show running-config ip pim dense-mode 7.46
show running-config ip pim sparse-mode 7.47
show running-config ip route 7.48
show running-config ipv6 access-list 7.49
show running-config ipv6 mroute 7.50

show running-config ipv6 prefix-list 7.51
show running-config ipv6 route 7.52
show running-config key chain 7.53
show running-config lldp 7.54
show running-config log 12.46
show running-config power-inline 7.55
show running-config prefix-list 7.56
show running-config route-map 7.57
show running-config router ipv6 vrrp 82.19
show running-config router vrrp 82.18
show running-config router-id 7.59
show running-config router 7.58
show running-config security-password 7.60
show running-config sflow 107.15
show running-config snmp 94.7
show running-config ssh 78.21
show running-config stack 111.13
show running-config trigger 103.9
show running-config 7.37
show security-password configuration 5.30
show security-password user 5.31
show sflow interface 107.17
show sflow 107.16
show snmp-server community 94.8
show snmp-server group 94.9
show snmp-server user 94.9
show snmp-server view 94.10
show snmp-server 94.8
show spanning-tree brief 21.16
show spanning-tree mst config 21.18
show spanning-tree mst detail interface 21.20
show spanning-tree mst detail interface 21.24
show spanning-tree mst detail 21.19
show spanning-tree mst instance interface 21.22
show spanning-tree mst instance 21.21
show spanning-tree mst interface 21.23
show spanning-tree mst 21.17
show spanning-tree statistics instance interface 21.27
show spanning-tree statistics instance 21.26
show spanning-tree statistics interface 21.28
show spanning-tree statistics 21.25
show spanning-tree vlan range-index 21.30
show spanning-tree 21.13
show ssh client 78.23
show ssh server allow-users 78.25
show ssh server deny-users 78.26
show ssh server 78.24
show ssh 78.22
show stack resiliencylink 111.20
show stack 111.15
show startup-config 7.61
show static-channel-group 23.19
show storm-control 17.49
show system environment 10.48
show system interrupts 10.49
show system mac 10.50
show system pci device 10.51

show system pci tree 10.52
show system pluggable detail 10.55
show system pluggable diagnostics 10.59
show system pluggable 10.53
show system serialnumber 10.62
show system 10.47
show tacacs+ 75.2
show tech-support 10.63
show telnet 5.32
show test cable-diagnostics tdr 109.2
show trigger 103.10
show users 5.33
show version 7.62
show vlan classifier group interface 19.11
show vlan classifier group 19.10
show vlan classifier interface group 19.12
show vlan classifier rule 19.13
show vlan private-vlan 19.14
show vlan statistics 19.15
show vlan 19.9
show vrrp counters 82.22
show vrrp ipv6 ..... 82.21
show vrrp (session) 82.24
show vrrp 82.20
show <epsr-instance> counters 84.20
shutdown 14.15
snmp trap link-status suppress 94.12
snmp trap link-status 94.11
snmp-server community 94.16
snmp-server contact 94.17
snmp-server enable trap 94.18
snmp-server engineID local reset 94.23
snmp-server engineID local 94.21
snmp-server group 94.24
snmp-server host 94.25
snmp-server location 94.27
snmp-server source-interface 94.28
snmp-server startup-trap-delay 94.29
snmp-server user 94.30
snmp-server view 94.32
snmp-server 94.14
sntp-address 92.40
source-ip 105.20
spanning-tree autoedge (RSTP and MSTP) 21.31
spanning-tree bpdu 21.32
spanning-tree cisco-interoperability (MSTP) 21.34
spanning-tree edgeport (RSTP and MSTP) 21.35
spanning-tree enable 21.36
spanning-tree errdisable-timeout enable 21.38
spanning-tree errdisable-timeout interval 21.39
spanning-tree force-version 21.40
spanning-tree forward-time 21.41
spanning-tree guard root 21.42
spanning-tree hello-time 21.43
spanning-tree link-type 21.44
spanning-tree max-age 21.45
spanning-tree max-hops (MSTP) 21.46

spanning-tree mode 21.47
spanning-tree mst configuration 21.48
spanning-tree mst instance path-cost 21.50
spanning-tree mst instance priority 21.52
spanning-tree mst instance restricted-role 21.53
spanning-tree mst instance restricted-tcn 21.54
spanning-tree mst instance 21.49
spanning-tree path-cost 21.55
spanning-tree portfast bpdu-filter 21.58
spanning-tree portfast bpdu-guard 21.60
spanning-tree portfast (STP) 21.56
spanning-tree priority (bridge priority) 21.62
spanning-tree priority (port priority) 21.63
spanning-tree restricted-role 21.64
spanning-tree restricted-tcn 21.64
spanning-tree transmit-holdcount 21.65
speed (asyn) 10.66
speed 17.50
ssh client 78.29
ssh server allow-users 78.33
ssh server authentication 78.35
ssh server deny-users 78.37
ssh server resolve-host 78.38
ssh server scp 78.39
ssh server sftp 78.40
ssh server 78.31
ssh 78.27
stack disabled-master-monitoring 111.21
stack enable 111.22
stack management subnet 111.23
stack management vlan 111.24
stack priority 111.25
stack renumber cascade 111.27
stack renumber 111.26
stack resiliencylink 111.28
stack software-auto-synchronize 111.30
stack virtual-chassis-id 111.31
stack virtual-mac 111.32
static-channel-group 23.20
storm-action 63.48
storm-control level 17.52
storm-downtime 63.49
storm-protection 63.50
storm-rate 63.51
storm-window 63.52
subnet-mask 90.54
summary-address (IPv6 OSPF) 44.79
summary-address 42.91
switch provision (stack) 111.33
switchport access vlan 19.16
switchport atmf-crosslink 86.88
switchport atmf-link 86.90
switchport enable vlan 19.17
switchport mode access 19.18
switchport mode private-vlan trunk promiscuous 19.20
switchport mode private-vlan trunk secondary 19.22
switchport mode private-vlan 19.19

switchport mode trunk 19.24
switchport port-security aging 17.54
switchport port-security maximum 17.55
switchport port-security violation 17.56
switchport port-security 17.53
switchport private-vlan host-association 19.25
switchport private-vlan mapping 19.26
switchport resiliencylink 111.34
switchport trunk allowed vlan 19.27
switchport trunk native vlan 19.30
switchport vlan-stacking (double tagging) 19.31
switchport voice dscp 19.32
switchport voice vlan priority 19.35
switchport voice vlan 19.33
system territory (deprecated) 10.67

T

tacacs-server host 75.3
tacacs-server key 75.5
tacacs-server timeout 75.6
tcpdump 29.54
telnet server 5.35
telnet 5.34
terminal length 5.36
terminal monitor 10.67
terminal resize 5.37
test cable-diagnostics tdr interface 109.3
test interface 15.4
test 103.15
thrash-limiting 17.57
time (trigger) 103.16
timeout (ping polling) 105.21
timers spf exp (IPv6 OSPF) 44.82
timers spf exp 42.92
timers spf (IPv6 OSPF) (deprecated) 44.81
timers (IPv6 RIPng) 40.23
timers (RIP) 38.48
traceroute ipv6 ..... 31.29
traceroute 29.55
transition-mode 82.25
trap 103.18
trigger activate 103.20
trigger 103.19
trust dscp 63.53
tunnel dscp (6to4) 33.3
tunnel mode ipv6ip 33.4
tunnel source 33.5
tunnel ttl 33.6
type atmf node 103.21
type atmf node 86.91
type cpu 103.23
type interface 103.24
type memory 103.25
type periodic 103.26
type ping-poll 103.27

type reboot....103.27
type stack disabled-master....103.28
type stack link....103.29
type stack master-fail....103.29
type stack member....103.30
type time....103.31

U

undebug aaa....67.29
undebug all ipv6 pim sparse-mode....53.53
undebug all pim dense-mode....55.25
undebug all pim sparse-mode....51.45
undebug all....10.67
undebug dot1x....69.29
undebug epsr....84.22
undebug igmp....49.51
undebug ipv6 ospf events....44.83
undebug ipv6 ospf ifsm....44.83
undebug ipv6 ospf lsa....44.83
undebug ipv6 ospf nfsm....44.83
undebug ipv6 ospf packet....44.83
undebug ipv6 ospf route....44.83
undebug ipv6 pim sparse-mode....53.54
undebug ipv6 rip....40.24
undebug lacp....23.21
undebug loopprot....17.58
undebug mail....98.7
undebug mstp....21.65
undebug ospf events....42.93
undebug ospf ifsm....42.93
undebug ospf lsa....42.93
undebug ospf nfsm....42.93
undebug ospf nsm....42.93
undebug ospf packet....42.93
undebug ospf route....42.93
undebug ping-poll....105.22
undebug platform packet....17.58
undebug radius....71.19
undebug rip....38.49
undebug sflow....107.17
undebug snmp....94.33
undebug ssh client....78.41
undebug ssh server....78.41
undebug stack....111.36
undebug trigger....103.31
undebug vrrp events....82.26
undebug vrrp packet....82.27
undebug vrrp....82.26
up-count....105.22
user (RADIUS server)....73.36
username....5.38

V

version 38.50
virtual-ipv6 ..... 82.30
virtual-ip 82.28
vlan classifier activate 19.37
vlan classifier group 19.38
vlan classifier rule ipv4 ..... 19.39
vlan classifier rule proto 19.40
vlan database 19.43
vlan mode stack-local-vlan 111.35
vlan mode stack-local-vlan 19.44
vlan statistics 19.45
vlan (RADIUS server) 73.38
vlan 19.36
vrrp vmac 82.32

W

wait 13.4
write file 7.63
write memory 7.63
write terminal 7.63
wrr-queue disable queues 63.54
wrr-queue egress-rate-limit queues 63.55
wrr-queue weight queues 63.56

Z

(access-list extended ICMP filter) 59.15
(access-list extended IP filter) 59.17
(access-list extended IP protocol filter) 59.20
(access-list extended TCP UDP filter) 59.24
(access-list hardware ICMP filter) 58.20
(access-list hardware IP protocol filter) 58.23
(access-list hardware MAC filter) 58.28
(access-list hardware TCP UDP filter) 58.31
(access-list standard named filter) 59.31
(access-list standard numbered filter) 59.33
(ipv6 access-list extended IP protocol filter) 61.11
(ipv6 access-list extended TCP UDP filter) 61.14
(ipv6 access-list named ICMP filter) 60.7
(ipv6 access-list named protocol filter) 60.10
(ipv6 access-list named TCP UDP filter) 60.14
(ipv6 access-list standard filter) 61.18

Appendix B: Changes in this Software Version

Introduction B.2
Key New Features and Enhancements in version 5.4.4-1.1 ..... B.3
- Cable Fault Locator B.3
- AMF Enhancements B.3

Important Considerations Before Upgrading to Version 5.4.4-1.1 ..... B.5
- Upgrading a VCStack B.5
- Forming or extending a VCStack B.5
- AMF software version compatibility B.5
- Upgrading all switches in an AMF network B.5

Key New Features and Enhancements in version 5.4.4-2.3 ..... B.7
- Web-Authentication Enhancements B.7
- Stack-Local-VLANs Enhancements B.9

Important Considerations Before Upgrading to Version 5.4.4-2.3 ..... B.11
- Upgrading a VCStack B.11
- Forming or extending a VCStack B.11
- AMF software version compatibility B.11
- Upgrading all switches in an AMF network B.12

New Features and Enhancements in Version 5.4.4-3.5 ..... B.13
- Login Security Enhancements B.13
- VRRPv3 Secondary IPv6 Address B.14
- Web Authentication Unmatched Proxy Setting Detection B.14

Important Considerations Before Upgrading to this Version B.15
- Upgrading a VCStack B.15
- Forming or extending a VCStack B.15
- AMF software version compatibility B.15
- Upgrading all switches in an AMF network B.16

Installing Software Version 5.4.4-3.5 ..... B.17
Installing the GUI B.19

Introduction

This appendix describes the new features and enhancements for x510 Series Switches in the following major and minor AlliedWare Plus software versions:

- 5.4.4-1.1
- 5.4.4-2.3
- 5.4.4-3.5

Key New Features and Enhancements in version 5.4.4-1.1

Software version 5.4.4-1.1 includes all the new features that have been added to AlliedWare Plus since the release of 5.4.4-0.1.

This section summarizes the key new features.

Cable Fault Locator

The Cable Fault Locator (CFL) is a cable diagnostic tool for copper (but not fiber) cables. You can select a port and the CFL will display, for that port, connection status or faults that exist in either the connected cable or in its terminations. The CFL operates using a technology known as Time Domain Reflectometry (TDR) to test all four pairs of wires inside the cable.

CFL is supported on the x510, x510L and x510-DP Series switches.

The following new commands support this feature:

clear test cable-diagnostics tdr	This command clears the results of a cable-diagnostics CFL test.
show test cable-diagnostics tdr	This command displays the results of a cable-diagnostics CFL test.
test cable-diagnostics tdr interface	This command initiates cable-diagnostics tests to twisted pair data cables in order to detect either correct, short, or open circuit terminations.

For more information, see **Chapter 108, Cable Fault Locator Introduction** and **Chapter 109, Cable Fault Locator Commands**.

AMF Enhancements

Allied Telesis Management Framework (AMF) is a sophisticated suite of management tools that provides a simplified approach to network management. Since its initial release in software version 5.4.3-1.4, AMF has been continually enhanced with features to increase its versatility. The latest enhancements are described below.

Backup to remote file server

You can now choose to store your switch's backup data on a remote backup server rather than on the Master node's external media. The server is used for both backup and recovery. Each AMF master can support up to two remote file servers, which are mounted on the Master's file system.

Recovery progress LED indication

This feature displays the recovery status during automatic recovery. Two distinct flash patterns indicate the different possible states during node recovery: "node recovery in progress" and "node recovery failed". You can use a new command (atmf recover led-off) during a recovery to turn off the progress indication and return the port LEDs to their normal running state.

Node provisioning

You can now pre-configure, or provision, a port for a future node before it is added to the network. A provisioned node can be created as a new unique entity, or can be cloned using the backup data from an existing node. When you add the new node to the provisioned port in the AMF network, its configuration is automatically loaded from the information stored in the backup media, with no further effort from you.

Node cleaning

A clean device is one that has had its previous release and configuration components removed. Thanks to the new **atmf cleanup** command you can now easily return a used switch to its original "out-of-the-box" state.

This process of cleaning is required when replacing a device with one that has been used previously and still retains components of its previous configuration. Once you have cleaned a switch, you can connect it to your AMF network and know that automatic node recovery will start effortlessly.

Important Considerations Before Upgrading to Version 5.4.4-1.1

Upgrading a VCStack

This software version supports VCStack "reboot rolling" upgrades. With the **reboot rolling** command, you can reduce downtime when upgrading a VCStack.

You can use the **reboot rolling** command to upgrade to 5.4.4-1.1 from any 5.4.4-0.x version.

However, if you want to use the **reboot rolling** command to upgrade from any 5.4.3-x.x version to 5.4.4-1.1, you must upgrade to 5.4.4-0.x first.

Forming or extending a VCStack

If you create a VCStack from switches that are running different software versions, auto-synchronization ensures that all members will run the same software version when they boot up.

However, auto-synchronization is not supported between all versions of 5.4.4. The following table lists compatible versions:

If the existing VCStack is running ...	then a new member can join the VCStack if it is running ...
any 5.4.4-0.x version	any 5.4.4-0.x version
5.4.4-1.1 or 5.4.4-1.2	5.4.4-1.1 or 5.4.4-1.2

Before you add a new switch to a stack, make sure the new switch's version is compatible with the stack's version. If the new switch is running an incompatible version, it cannot join the stack until you have manually upgraded it.

AMF software version compatibility

We strongly recommend that all switches in an AMF network run the same software release.

If this is not possible, switches running this minor version are compatible with switches running version 5.4.3-2.6 and later, or any 5.4.4-x.x version.

Upgrading all switches in an AMF network

This version supports upgrades across AMF networks. There are two methods for upgrading firmware on an AMF network:

- Reboot-rolling, which upgrades and reboots each switch in turn
- Distribute firmware, which upgrades each switch, but does not reboot them. This lets you reboot the switches at a minimally-disruptive time.

You can use either of these methods to upgrade to this minor software version from 5.4.3-2.6 and later, or from any 5.4.4-0.x version.

In summary, the process for upgrading firmware on an AMF network is:

1. Copy the release .rel files for each switch family to the media location you intend to upgrade from (Flash memory, SD card, USB stick etc).
2. Decide which AMF upgrade method is most suitable.
3. Initiate the AMF network upgrade using the selected method. To do this:
   a. create a working-set of the switches you want to upgrade
   b. enter the command **atmf reboot-rolling <*location*>** or **atmf distribute-firmware <*location*>** where **<*location*>** is the location of the .rel files.
   c. Check the console messages to make sure that all switches are “release ready”. If they are, follow the prompts to perform the upgrade.

Key New Features and Enhancements in version 5.4.4-2.3

Software version 5.4.4-2.3 includes all the new features that have been added to AlliedWare Plus since the release of 5.4.4-1.1.

This section summarizes the key new features.

Web-Authentication Enhancements

The following enhancements have been added to web-authentication.

- Custom login page
- External login page
- Robust web-authentication

Custom login page

You can customize the web-authentication page by changing the web page logo image, success message, welcome message, and web page title.

The following commands have been introduced for this enhancement.

- **auth-web-server page logo**
- **auth-web-server page sub-title**
- **auth-web-server page success-message**
- **auth-web-server page title**
- **auth-web-server page welcome-message**
- **show auth-web-server page**

External login page

You can use an external login page for web-authentication rather than using the built-in AlliedWare Plus login page.

The **auth-web forward** command has been introduced for this enhancement.

Robust web-authentication

Web-authentication configuration has been simplified and some limitations have been removed. For command details, see **Chapter 65, Authentication Commands**.

- Previously, you could configure an intercept mode on the web-authentication server for supplicants (client devices). Now, you no longer need to configure the intercept mode. Intercept mode is always available and it intercepts HTTP packets but doesn't intercept ARP or DNS messages. As a result, the **auth-web-server mode** command has been deleted.
- Previously, you could enable the HTTP redirect feature on every interface on which web-based port authentication was enabled. Now, the HTTP redirect feature is always enabled and you cannot disable it. As a result, the **auth-web-server http-redirect** command has been deleted.
- Previously, you needed to register the gateway information when the supplicant was authorized. Now, the AlliedWare Plus device acts as the default gateway and you no longer need to add the gateway information. As a result, the **auth-web-server gateway** command has been deleted.
- Previously, you could set the HTTPS port number for the web authentication server. Now, you no longer need to set the port number and the default port number 443 is used. As a result, the **auth-web-server sslport** command has been deleted.

- The default behavior of web-authentication packet forwarding has changed. Previously, packet forwarding for port authentication was disabled by default. Now, ARP, DHCP, DNS forwarding for port authentication are enabled by default. TCP and UDP forwarding for port authentication are disabled by default. As a result, the default behavior of the **auth-web forward** command has been changed.
- Previously, you could use either HTTP protocol or HTTPS protocol for the web authentication server. Both HTTP and HTTPS packets were redirected to HTTP server or HTTPS server. Now, you can use both HTTP protocol and HTTPS protocol. When both protocols are used, HTTP packet is redirected to HTTP server and HTTPS packet is redirected to HTTPS server respectively. As a result, the **auth-web-server ssl** command has been changed and you can use the **hybrid** option of this command to enable both HTTP and HTTPS for the web authentication server.
- Previously, you could register only HTTP intercept port numbers. Now, you can use the **auth-web-server ssl intercept-port** new command to register HTTPS intercept port numbers when the HTTPS server uses custom port numbers.
- Previously, you couldn't assign a hostname to the web authentication server. Now, you can use the **auth-web-server host-name** new command to assign a hostname to the web authentication server.
- As a result of the enhancements, the output of the **show auth-web-server** command has been changed.
- If you configure a virtual IP address for the web-authentication server by using the **auth-web-server ipaddress** command or the **auth-web-server dhcp ipaddress** command, you must add a hardware ACL which sends the packets going to the virtual IP address to the CPU on the web-authentication enabled interfaces. If the hardware ACL is not set, the web-authentication success page will not appear on the supplicant's web browser. For example, if you configure the virtual IP address 1.2.3.4 and web-authentication is enabled on port1.0.1 and port1.0.7, you must add the hardware filter **send-to-cpu ip any 1.2.3.4/32** to port1.0.1 and port1.0.7 as shown in the following **show running-config** command output:

```
…
auth-web-server ipaddress 1.2.3.4

access-list hardware acl-web
send-to-cpu ip any 1.2.3.4/32
!
interface port1.0.1
auth-web enable
access-group acl-web
!
interface port1.0.7
auth-web enable
access-group acl-web
!
```

Stack-Local-VLANs Enhancements

Network data VLANs are shared by the stack and use the stack's virtual MAC address. Consequently only the stack master is able to respond to messages such as ARP or ICMP requests. One disadvantage of this is that although network administrators can ping the whole stack to determine its operational status, such pinging will not provide status information for individual stack members. Stack-local-VLANs provide a solution to this problem. For command details, see **vlan mode stack-local-vlan**.

Note This enhancement exists only on the following stackable switches: x310, x510, and x610 Series.

Stack-Local-VLAN Operation

Each stack-local-VLAN belongs to a specific stack member, and uses that stack member's physical MAC address, rather than the stack's virtual MAC address. This enables a stack member to process stack-local-VLAN traffic directly on its own CPU, even if this is the stack master.

This strict association of local VLAN, to specific stack member enables network administrators to ping each stack member individually in order to monitor the health of the entire stack, on a member-by-member basis.

Stack-local-VLANs are especially useful within networks where ping polling is used to monitor the health of network devices.

Stack-Local-VLAN Configuration

The following example shows a stack-local-VLAN configuration for a two member stack. Note that overlapping IP subnets are permitted on local VLAN interfaces:

Table 1-1: Configuring Stack-Local-VLANs on a Two Member Stack

Description	Prompt	Command
Step 1. Create the stack-local-VLANs for stack members 1 and 2		
Enter global configuration mode.	`awplus#`	`configure terminal`
Enter VLAN database mode.	`awplus(config)#`	`vlan database`
Create the stack-local-VLAN for stack member 1.	`awplus(config-vlan)#`	`vlan 4001 mode stack-local-vlan 1`
Create the stack-local-VLAN for stack member 2.	`awplus(config-vlan)#`	`vlan 4001 mode stack-local-vlan 2`
Step 2. Apply the access port mode to port 1.0.24		
Enter global configuration mode.	`awplus#`	`configure terminal`
Enter interface configuration mode for port 1.0.24.	`awplus(config)#`	`interface port1.0.24`
Set the port to access mode.	`awplus(config-if)#`	`switchport mode access`
Add this port to member 1's local VLAN.	`awplus(config-if)#`	`switchport access vlan 4001`
Step 3. Apply the access port mode to port 2.0.24		

Table 1-1: Configuring Stack-Local-VLANs on a Two Member Stack

Description (cont.)	Prompt (cont.)	Command (cont.)
Enter interface configuration mode for port 2.0.24.	`awplus(config)#`	`interface port2.0.24`
Set the port to access mode.	`awplus(config-if)#`	`switchport mode access`
Add this port to member 2's local VLAN.	`awplus(config-if)#`	`switchport access vlan 4002`
Step 4. Apply the IP address 192.168.1.1/24 to VLAN 4001		
Enter global configuration mode.	`awplus#`	`configure terminal`
Select local VLAN interface for member 1	`awplus(config)#`	`interface vlan4001`
Assign an IP address that member 1 will reply to.	`awplus(config-if)#`	`ip address 192.168.1.1/24`
Step 5. Apply the IP address 192.168.1.2/24 to VLAN 4002		
Enter global configuration mode.	`awplus#`	`configure terminal`
Select local VLAN interface for member 2	`awplus(config)#`	`interface vlan4002`
Assign an IP address that member 2 will reply to.	`awplus(config-if)#`	`ip address 192.168.1.2/24`

Important Considerations Before Upgrading to Version 5.4.4-2.3

Upgrading a VCStack

This software version supports VCStack "reboot rolling" upgrades. With the **reboot rolling** command, you can reduce downtime when upgrading a VCStack.

You can use the **reboot rolling** command to upgrade to 5.4.4-2.3 from any 5.4.4-1.x version. The following table shows the process for using it to update from earlier versions.

Upgrading from	How to upgrade using the reboot rolling command
5.4.4-0.x	First upgrade to 5.4.4-1.x, then to 5.4.4-2.x.
5.4.3-x.x	First upgrade to any 5.4.4-0.x version, then to 5.4.4-1.x, then to 5.4.4-2.x.

Forming or extending a VCStack

If you create a VCStack from switches that are running different software versions, auto-synchronization ensures that all members will run the same software version when they boot up.

However, auto-synchronization is not supported between all versions of 5.4.4. The following table lists compatible versions:

If the existing VCStack is running ...	then a new member can join the VCStack if it is running ...
any 5.4.4-0.x version	any 5.4.4-0.x version
5.4.4-1.1 or 5.4.4-1.2	5.4.4-1.1 or 5.4.4-1.2
5.4.4-2.3 or 5.4.4-2.4	5.4.4-2.3 or 5.4.4-2.4

Before you add a new switch to a stack, make sure the new switch's version is compatible with the stack's version. If the new switch is running an incompatible version, it cannot join the stack until you have manually upgraded it.

AMF software version compatibility

We strongly recommend that all switches in an AMF network run the same software release.

If this is not possible, switches running this minor version are compatible with switches running version 5.4.3-2.6 and later or any 5.4.4-x.x version.

Upgrading all switches in an AMF network

This version supports upgrades across AMF networks. There are two methods for upgrading firmware on an AMF network:

- Reboot-rolling, which upgrades and reboots each switch in turn
- Distribute firmware, which upgrades each switch, but does not reboot them. This lets you reboot the switches at a minimally-disruptive time.

You can use either of these methods to upgrade to this minor software version from 5.4.3-2.6 and later, or from any 5.4.4-x.x version.

In summary, the process for upgrading firmware on an AMF network is:

1. Copy the release .rel files for each switch family to the media location you intend to upgrade from (Flash memory, SD card, USB stick etc).

2. Decide which AMF upgrade method is most suitable.

3. Initiate the AMF network upgrade using the selected method. To do this:

   a. create a working-set of the switches you want to upgrade

   b. enter the command **atmf reboot-rolling <*location*>** or **atmf distribute-firmware <*location*>** where **<*location*>** is the location of the .rel files.

   c. Check the console messages to make sure that all switches are “release ready”. If they are, follow the prompts to perform the upgrade.

New Features and Enhancements in Version 5.4.4-3.5

Software version 5.4.4-3.5 includes all the new features that have been added to AlliedWare Plus since the release of 5.4.4-1.1.

This section summarizes the new features in 5.4.4-3.5.

Login Security Enhancements

This software version includes several enhancements to the switch's login security settings.

As indicated below, some of these enhancements are only available when the switch is "locked down" at security level 3. This security level is a bootloader security setting. To set it, type Ctrl-B during boot-up to enter the bootloader, then type "s" at the bootloader menu.

The bootloader security settings may not be available on all x510 Series switches.

"Last login" message

At login, the switch now displays:

- a "Last login" message, indicating when that user last logged in, and
- if the switch's bootloader is set to security level 3, a "Failed login" message if there have been any failed login attempts for that user.

These messages are displayed for logins via the console, Telnet or SSH.

The console output looks like this:

```
x510-D login: aa
Password:
Last login: Mon Oct 13 14:07:32 NZST 2014 on ttyS0
Last failed login: Mon Oct 13 15:21:07 NZST 2014 on ttyS0
There were 2 failed login attempts since the last successful
login.

AlliedWare Plus (TM) 5.4.4 10/13/14 12:59:36
```

Logging of attempts to set release files

When a user attempts to set a primary or backup release file (using the **boot system** command), the switch now sends a failure or success message to the logging system. The error message has a log severity level of "warning" and the success message has a severity level of "notice".

Delay between password attempts

On a switch whose bootloader is set to security level 3, a 4 second gap is now required between attempts to re-enter a password. This applies for console, Telnet and SSH logins, and for both local and RADIUS users.

Configurable number of login attempts by SSH

You can now specify the maximum number of SSH authentication attempts that the switch will allow. The default is 6 attempts. To change this, use the new command:

```
awplus(config)#ssh server max-auth-tries <1-32>
```

VRRPv3 Secondary IPv6 Address

VRRPv3 now allows users to specify a secondary IPv6 address on an IPv6 VRRP instance. This enables you to specify a globally-routable address as the default gateway address for all the hosts on a VLAN.

To do this, use the new **secondary** parameter in the following command:

Syntax

```
virtual-ipv6 <ipv6-address> [master|backup] [primary|
secondary]

no virtual-ipv6
```

Parameter	Description
<ipv6-address>	The IPv6 address of the virtual router, entered in hexadecimal, in the format X:X::X.X. This is an IPv6 link-local address.
`master`	Sets the default state of the VRRPv3 router within the Virtual Router as **master**. For **master**, the router must own the Virtual IP address.
`backup`	Sets the default state of the VRRPv3 router within the Virtual Router as **backup**.
`primary`	Sets the specified address as the primary IPv6 address. The primary address must be a link-local IPv6 address.
`secondary`	Sets the specified address as the secondary IPv6 address. Normally this would be a globally-routable IPv6 address.

Mode Router Configuration

Web Authentication Unmatched Proxy Setting Detection

By default, AlliedWare Plus Web Authentication intercepts the supplicant's initial TCP port 80 connection to a web page and sends it the Web Authentication login page. You can also specify any additional TCP port numbers that the web authentication server is to intercept by using the **auth-web-server intercept-port** command. In this way, Web Authentication can intercept packets going to a proxy server by adding the port number of the proxy server.

However, when the web authentication switch is in a guest network, the switch does not know the proxy server's port number in the supplicant's proxy setting. To overcome this limitation, you can now use the new **any** option in the **auth-web-server intercept-port** command to intercept all TCP packets:

```
awplus(config)#auth-web-server intercept-port any
```

Important Considerations Before Upgrading to this Version

Upgrading a VCStack

This software version supports VCStack "reboot rolling" upgrades. With the **reboot rolling** command, you can reduce downtime when upgrading a VCStack.

You can use the **reboot rolling** command to upgrade to 5.4.4-3.5 from any 5.4.4-1.x version. The following table shows the process for using it to update from earlier versions.

Upgrading from	How to upgrade using the reboot rolling command
5.4.4-0.x	First upgrade to 5.4.4-1.x, then to 5.4.4-2.x.
5.4.3-x.x	First upgrade to any 5.4.4-0.x version, then to 5.4.4-1.x, then to 5.4.4-2.x.

Forming or extending a VCStack

If you create a VCStack from switches that are running different software versions, auto-synchronization ensures that all members will run the same software version when they boot up.

However, auto-synchronization is not supported between all versions of 5.4.4. The following table lists compatible versions:

If the existing VCStack is running ...	then a new member can join the VCStack if it is running ...
any 5.4.4-0.x version	any 5.4.4-0.x version
5.4.4-1.1 or 5.4.4-1.2	5.4.4-1.1 or 5.4.4-1.2
5.4.4-2.3 or 5.4.4-2.4	5.4.4-2.3 or 5.4.4-2.4
5.4.4-3.5	5.4.4-2.3, 5.4.4-2.4 or 5.4.4-3.5

Before you add a new switch to a stack, make sure the new switch's version is compatible with the stack's version. If the new switch is running an incompatible version, it cannot join the stack until you have manually upgraded it.

AMF software version compatibility

We strongly recommend that all switches in an AMF network run the same software release.

If this is not possible, switches running this minor version are compatible with switches running version 5.4.3-2.6 and later or any 5.4.4-x.x version.

Upgrading all switches in an AMF network

This version supports upgrades across AMF networks. There are two methods for upgrading firmware on an AMF network:

- Reboot-rolling, which upgrades and reboots each switch in turn
- Distribute firmware, which upgrades each switch, but does not reboot them. This lets you reboot the switches at a minimally-disruptive time.

You can use either of these methods to upgrade to this minor software version from 5.4.3-2.6 and later, or from any 5.4.4-x.x version.

In summary, the process for upgrading firmware on an AMF network is:

1. Copy the release .rel files for each switch family to the media location you intend to upgrade from (Flash memory, SD card, USB stick etc).

2. Decide which AMF upgrade method is most suitable.

3. Initiate the AMF network upgrade using the selected method. To do this:

   a. create a working-set of the switches you want to upgrade

   b. enter the command **atmf reboot-rolling <*location*>** or **atmf distribute-firmware <*location*>** where **<*location*>** is the location of the .rel files.

   c. Check the console messages to make sure that all switches are “release ready”. If they are, follow the prompts to perform the upgrade.

Installing Software Version 5.4.4-3.5

To install and enable this software version, use the following steps:

1. Copy the software version file (.rel) onto your TFTP server.

2. If necessary, delete or move files to create space in the switch's Flash memory for the new file. To see the memory usage, use the command:

   ```
   awplus# show file systems
   ```

   To list files, use the command:

   ```
   awplus# dir
   ```

   To delete files, use the command:

   ```
   awplus# del <filename>
   ```

   You cannot delete the current boot file.

3. Copy the new release from your TFTP server onto the switch.

   ```
   awplus# copy tftp flash
   ```

   Follow the onscreen prompts to specify the server and file.

4. Move from Privileged Exec mode to Global Configuration mode, using:

   ```
   awplus# configure terminal
   ```

   Then set the switch to reboot with the new software version:

Switch	Command
x210 Series	`awplus(config)# boot system x210-5.4.4-3.5.rel`
x230 Series	`awplus(config)# boot system x230-5.4.4-3.5.rel`
x310 Series	`awplus(config)# boot system x310-5.4.4-3.5.rel`
IX5-28GPX	`awplus(config)# boot system IX5-5.4.4-3.5.rel`
x510 Series	`awplus (config)#boot system x510-5.4.4-3.5.rel`
x610 Series	`awplus(config)# boot system x610-5.4.4-3.5.rel`
x900 Series	`awplus(config)# boot system x900-5.4.4-3.5.rel`
SBx908	`awplus(config)# boot system SBx908-5.4.4-3.5.rel`
SBx8100 with CFC400	`awplus(config)# boot system SBx81CFC400-5.4.4-3.5.rel`
SBx8100 with CFC960	`awplus(config)# boot system SBx81CFC960-5.4.4-3.5.rel`

Return to Privileged Exec mode and check the boot settings, by using the commands:

```
awplus(config)# exit
awplus# show boot
```

5. Reboot using the new software version.

```
awplus# reload
```

Installing the GUI

This section describes how to install and set up the AlliedWare Plus GUI using an SD card, a USB storage device, or a TFTP server. The version number in the GUI Java applet filename (**.jar)** gives the earliest version of the software file (**.rel**) that the GUI can operate with.

To install and run the AlliedWare Plus GUI requires the following system products and setup:

- PC Platform:
  Windows XP SP2 and up / Windows Vista SP1 and up
- Browser: (must support Java Runtime Environment (JRE) version 6)
  Microsoft Internet Explorer 7.0 and up / Mozilla Firefox 2.0 and up

To install the GUI on your switch, use the following steps:

1. Copy to the GUI Java applet file (**.jar** extension) onto your TFTP server, SD card or USB storage device.

2. Connect to the switch's management port, then log into the switch.

3. If necessary, delete or move files to create space in the switch's Flash memory for the new file.

   To see the memory usage, use the command:

   ```
   awplus# show file systems
   ```

   To list files, use the command:

   ```
   awplus# dir
   ```

   To delete files, use the command:

   ```
   awplus# del <filename>
   ```

   You cannot delete the current boot file.

4. Assign an IP address for connecting to the GUI. Use the commands:

   ```
   awplus# configure terminal
   awplus(config)# interface vlan1
   awplus(config-if)#ip address <address>/<prefix-length>
   ```

   Where <*address*> is the IP address that you will subsequently browse to when you connect to the GUI Java applet. For example, to give the switch an IP address of 192.168.2.6, with a subnet mask of 255.255.255.0, use the command:

   ```
   awplus(config-if)# ip address 192.168.2.6/24
   ```

5. If required, **configure a default gateway for the switch.**

   ```
   awplus(config-if)# exit
   awplus(config)# ip route 0.0.0.0/0 <gateway-address>
   ```

   Where <*gateway-address*> is the IP address for your gateway device. You do not need to define a default gateway if you browse to the switch from within its own subnet.

6. Copy the GUI file onto your switch from the TFTP server, SD card, or USB storage device.

   TFTP server: Use the command:

   ```
   awplus# copy tftp://<server-address>/<filename.jar> flash:/
   ```

   SD card: use the command:

   ```
   awplus# copy card:/<filename.jar> flash:/
   ```

   USB storage device: use the command:

   ```
   awplus# copy usb:/<filename.jar> flash:/
   ```

   where <*server-address*> is the IP address of the TFTP server, and where <*filename.jar*> is the filename of the GUI Java applet.

7. Ensure the HTTP service is enabled on your switch. Use the commands:

   ```
   awplus# configure terminal
   awplus(config)# service http
   ```

   The HTTP service needs to be enabled on the switch before it accepts connections from a web browser. The HTTP service is enabled by default. However, if the HTTP has been disabled then you must enable the HTTP service again.

8. Create a user account for logging into the GUI.

   ```
   awplus(config)# username <username> privilege 15 password
                     <password>
   ```

   You can create multiple users to log into the GUI. For information about the **username** command, see the AlliedWare Plus Software Reference.

9. Start the Java Control Panel, to enable Java within a browser

   On your PC, start the Java Control Panel by opening the Windows Control Panel from the Windows Start menu. Then enter Java Control Panel in the search field to display and open the Java Control Panel.

   Next, click on the 'Security' tab. Ensure the 'Enable Java content in the browser' checkbox is selected on this tab.

10. Enter the URL in the Java Control Panel Exception Site List

    Click on the 'Edit Site List' button in the Java Control Panel dialog Security tab to enter a URL in the Exception Site List dialog. In the 'Exception Site List' dialog, enter the IP address you configured in Step 4, with a http:// prefix.

    After entering the URL click the Add button then click OK.

11. Log into the GUI.

    Start a browser and enter the switch's IP address. The GUI starts up and displays a login screen. Log in with the username and password specified in the previous step.

Appendix C: GUI Reference

Introduction C.3
Install the GUI and set the switch C.4
System Requirements C.4
Install the GUI to your switch using a USB storage device C.4
Install the GUI to your switch via TFTP server C.5
Set up your switch and log into the GUI C.6
Using the GUI C.10
System > Status C.10
System > Status > System Details C.13
System > Status > System Date and Time C.13
System > Status > Top Ten Utilised Ports C.14
System > Identity C.15
System > Environment Monitoring C.17
System > File Management C.18
System > File Management > Copy File C.21
System > File Management > Move File C.21
System > File Management > Download File C.22
System > File Management > Upload File C.23
System > File Management > Delete File C.24
System > File Management > Delete Folder C.24
System > Stacking C.25
System > Stacking > Configure Stacking C.28
System > Stacking > Configure Stack Member C.29
System > License Management C.30
System > License Management > Add Feature License C.31
System > License Management > Delete Feature License C.32
Switching > Ports C.33
Switching > Ports > Monitor Port C.36
Switching > Ports > Configure Port C.37
Switching > VLANs C.38
Switching > VLANs > Add VLAN C.40
Switching > Link Aggregation C.41
Switching > Link Aggregation > Add Static Channel C.44
Switching > Link Aggregation > Add Dynamic Channel C.45
Switching > FDB Table C.46
Switching > Power over Ethernet C.47
Switching > Power over Ethernet > Configure PSE C.49
Switching > Power over Ethernet > Configure Port C.50
IP > IP Interfaces C.52
IP > IP Interfaces > Configure Primary IP Address C.53
IP > IP Interfaces > Add Secondary IP Address C.54
IP > Static Routes C.55
IP > Static Routes > Add Static Route C.56
IP > ARP C.57
IP > DNS C.59
IP > DNS > Add DNS Server C.60
IP > IGMP Snooping C.61
IP > IGMP Snooping > Configure Interface C.62
Resiliency and High Availability > STP C.63

Resiliency and High Availability > EPSR C.66
Management > Device Utilities C.68
Management > NTP C.69
Management > NTP > Add NTP Association C.71
Management > Remote CLI Access C.72
Management > Remote CLI Access > Telnet Settings C.74
Management > Remote CLI Access > SSH Settings C.74
Management > Logs C.75
Management > Logs > Export Logs C.76

Introduction

This appendix describes how to install, configure and use the Graphical User Interface (GUI) on switches running the AlliedWare Plus™ OS. The GUI provides extensive monitoring and essential configuration functionality for Allied Telesis switches via a web browser. This document explains how to install the GUI using either a USB storage device or via a TFTP server.

The GUI functionality is provided via a Java applet file. Before you can use the GUI to manage your switches, you must download the Java applet file, and install it to your switch's Flash file system.

Once the Java applet file is present in your switch's Flash, no specific commands are required to enable the GUI, or to inform the switch which Java applet file to use. Instead, when an incoming browser connection is established with the switch, the switch will automatically send the most recent compatible Java applet file that is present in its Flash file system.

Different versions of the Java applet file will be compatible with different versions of the AlliedWare Plus™ OS. The AlliedWare Plus™ OS automatically determines if a Java applet file is compatible, so the Java applet file that is delivered to your browser will always be compatible with the AlliedWare Plus™ OS version running on the switch to which you have connected.

Note which products and software version the GUI works with, along with PC and browser specifications listed. You may need to install and run the latest Java Runtime Environment that you can download from the Sun site so your browser can fully support the GUI Java applet.

Install the GUI and set the switch

This section shows you how to install and setup the AlliedWare Plus™ GUI on your switch.

System Requirements

To install and run the AlliedWare Plus™ GUI you will require the following system products and setup:

- PC Platform:
  Windows XP SP2 and up / Windows Vista SP1 and up
- Browser: (must support Java Runtime Environment (JRE) version 6)
  Microsoft Internet Explorer 7.0 and up / Mozilla Firefox 2.0 and up

Install the GUI to your switch using a USB storage device

Step 1: Download a GUI Java applet

AlliedWare Plus™ Operating System software from the Software Download area of the Allied Telesis Website: **www.alliedtelesis.com/support/software/restricted.** Login using your assigned Email Address and Password.

Download the Java applet file. This file will have a `.zip` file name extension. You need to extract the Java `.jar` file from the compressed `.zip` file. The version number of the software applet file (`.jar`) gives the earliest version of the software file (`.rel`) that the GUI can operate with.

Step 2: Copy the GUI Java applet .jar file on a USB storage device to flash memory

Insert the USB storage device into the switch. Connect to the management port, then log into the switch.

Copy the GUI Java applet to your switch, using the below commands:

```
awplus# copy usb:/<filename.jar> flash:/
```

Where `<filename.jar>` is the GUI Java applet file you downloaded in Step 1.

Note

Where the GUI file is not in the root directory of the USB flash drive, you must enter the full path to the GUI file. For example, where the GUI file resided in the folder gui_files, you would enter the command:
copy usb:/gui_files/filename.jar flash:/

Install the GUI to your switch via TFTP server

Step 1: Download a GUI Java applet file from the support site:

The GUI Java applet file is available in a compressed (`.zip`) file with the AlliedWare Plus™ Operating System software from the Support area of the Allied Telesis Website: **www.alliedtelesis.com**. Download the Java applet file. This file will have a `.zip` file name extension. You need to extract the Java `.jar` file from the compressed `.zip` file. The version number of the software applet file (.jar) gives the earliest version of the software file (`.rel`) that the GUI can operate with.

Step 2: Copy the GUI applet

Copy the GUI applet **.jar** file onto a TFTP server. Ensure this TFTP server is enabled and ready for the switch. Connect to the management port of the switch, then login to the switch. Do not connect to the management port of the TFTP server

Step 3: Copy the GUI Java applet to your switch

Use the following commands to copy the GUI Java applet to your switch:

```
awplus# copy tftp://<server-address>/
        <filename.jar> flash:/
```

where *<server-address>* is the IP address for the TFTP server, and where *<filename.jar>* is the GUI Java applet file you downloaded in Step 1.

Set up your switch and log into the GUI

Step 1: Assign the IP addresses:

Use the following commands to configure your switch with an appropriate IP address:

```
awplus# configure terminal
awplus(config)# interface vlan1
awplus(config-if)# ip address <address>/<prefix-
length>
```

where *<address>* is the IP address that you will subsequently browse to when you connect to the GUI Java applet. For example, to give the switch an IP address of `192.168.2.6`, and a subnet mask of `255.255.255.0`, use the following command:

```
awplus(config-if)# ip address 192.168.2.6/24
```

Step 2: Configure the Default Gateway

If necessary, use the following commands to configure the default gateway.

```
awplus(config-if)# exit
awplus(config)# ip route 0.0.0.0/0 <gateway-address>
```

where *<gateway-address>* is the IP address for your gateway device. Note that you do not need to define a default gateway if you browse to the switch from within its own subnet.

Step 3: Create a user account

In order to log into the GUI, you must first create a user account. Use the following commands to set up a user account

```
awplus(config)# username <username> privilege 15
password <password>
```

Note that you can create multiple users to log into the GUI. See the AlliedWare Plus Software Reference for information about the **username** command. The switch must be configured with a local database user, or the switch must be configured to remotely authenticate users with either TACACS+ or RADIUS.

Step 4: Ensure HTTP service is enabled

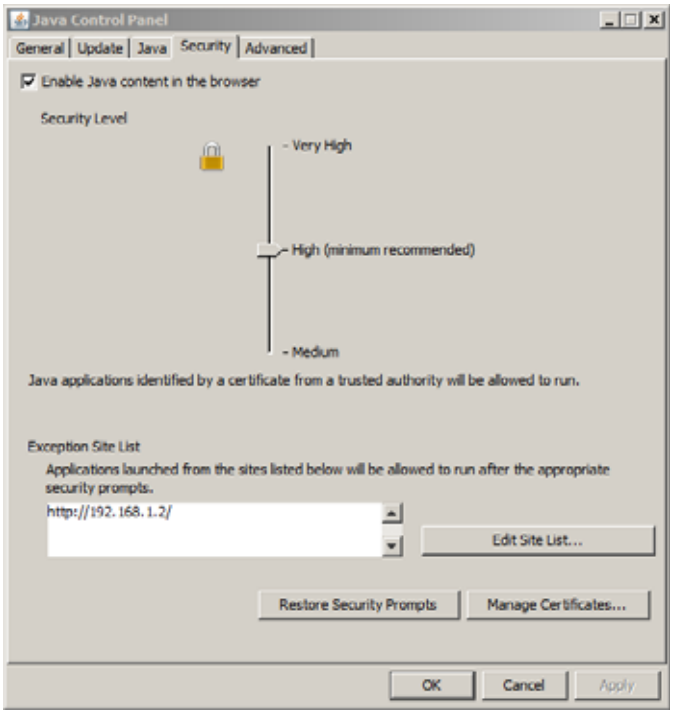
The HTTP service needs to be enabled on the switch before it accepts connections from a web browser. The HTTP service is enabled by default. However, if the HTTP service has been disabled then you must enable the HTTP service again. If the HTTP service is disabled then use the following command to enable the HTTP service:

```
awplus(config)# service http
```

See the AlliedWare Plus™ Software Reference for information about the **service http** command.

Step 5: Start the Java Control Panel to enable Java within a browser

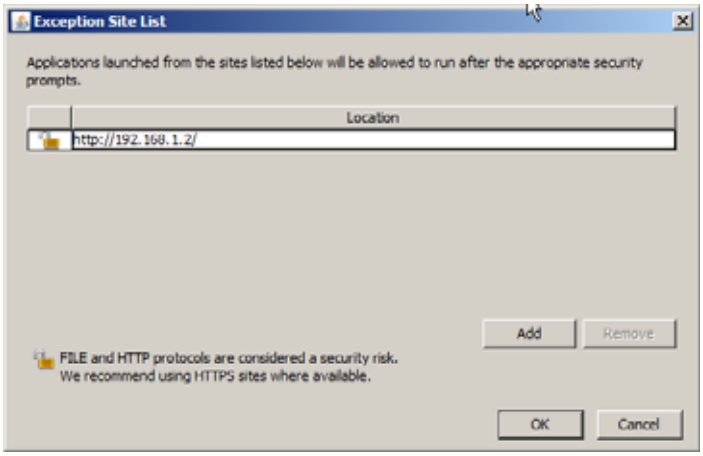
On your PC, start the 'Java Control Panel' by opening the 'Windows Control Panel' from the' Windows Start menu'. Then enter 'Java Control Panel' in the 'search' field to display and open the 'Java Control Panel'. Next, click on the 'Security' tab as below:



Ensure the 'Enable Java content in the browser' checkbox is selected on this tab.

Step 6: Enter the URL in the Java Control Panel Exception Site List

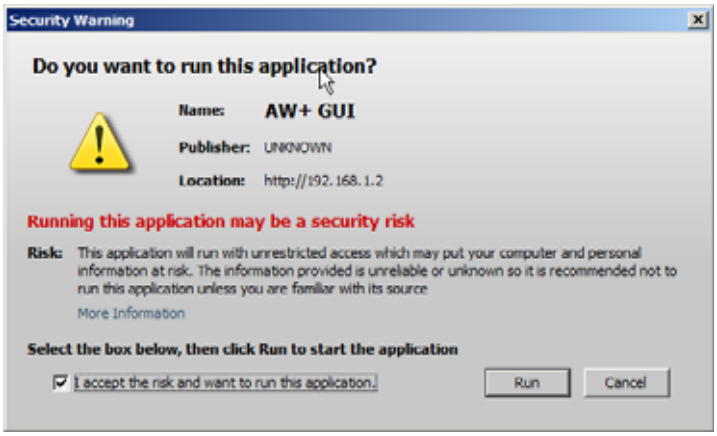
Click on the 'Edit Site List' button in the 'Java Control Panel' dialog 'Security' tab to enter a URL in the below 'Exception Site List dialog'. Enter the IP address you configured in Step 1 with an http:// prefix in the 'Exception Site List' dialog below:



After entering the URL click on the 'Add' button then click OK to start the GUI next.

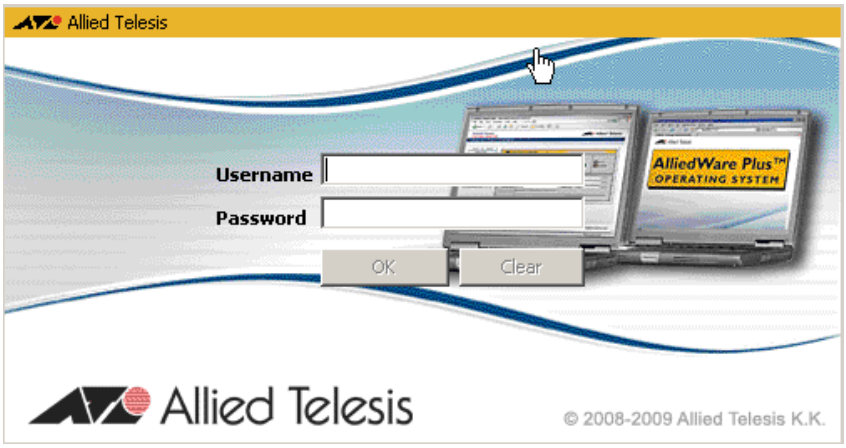
Step 7: Starting the GUI in a browser

Start a browser then enter the IP address you configured in Step 1 as the URL. You will then be presented with a 'Security Warning' before you can run the GUI Java applet. Click on the checkbox and click on 'Run 'to continue to log into the GUI applet:



Step 8: Log into the GUI

You will then be presented with a login screen after the GUI Java applet has started. You can then log in with the username and password that you defined in Step 3.



Step 9: Security Check

You may also be presented with a security check password prompt. This will occur when you have logged into the GUI with a user that is configured on the switch and has a privilege level of less than 15, or if the switch has been configured to authenticate enable passwords via TACACS+ using the **aaa authentication enable default group tacacs+** command.

You must enter the privilege level 15 enable password configured on the switch to access GUI configuration dialogs. If you enter an incorrect enable password, or no privilege level 15 enable password has been configured, then a message is shown stating you can use the GUI to monitor the switch, but not to configure the switch.

Using the GUI

This section explains how to use the AlliedWare Plus™ GUI. It assumes that you have installed the GUI on your switches and have the setup the browser on your PC. This procedure is covered in **"Install the GUI and set the switch" on page 1.4**.

In this section each screen is presented by its tab name and explains the content of the screen components.

System > Status

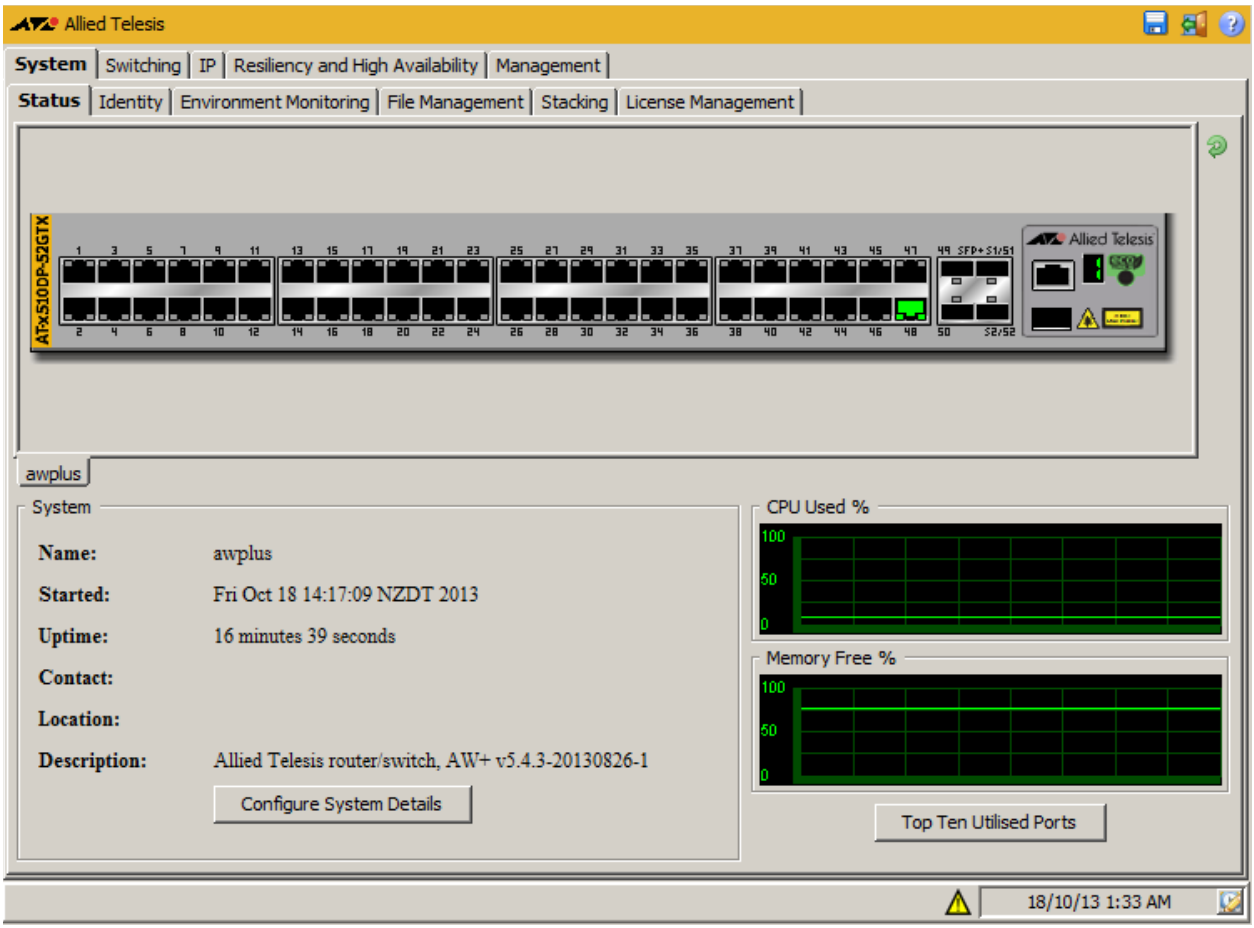
The **System** > **Status** menu tab enables you to display and configure basic system information.

The **CPU Used %** and **Memory Free %** graphs provide a brief history of CPU and memory usage.

Note

For systems equipped and configured using VCStack, there is a separate tab for each stack member with the system name displayed on each tab.
The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Example showing the **System** > **Status** menu tab:



Description

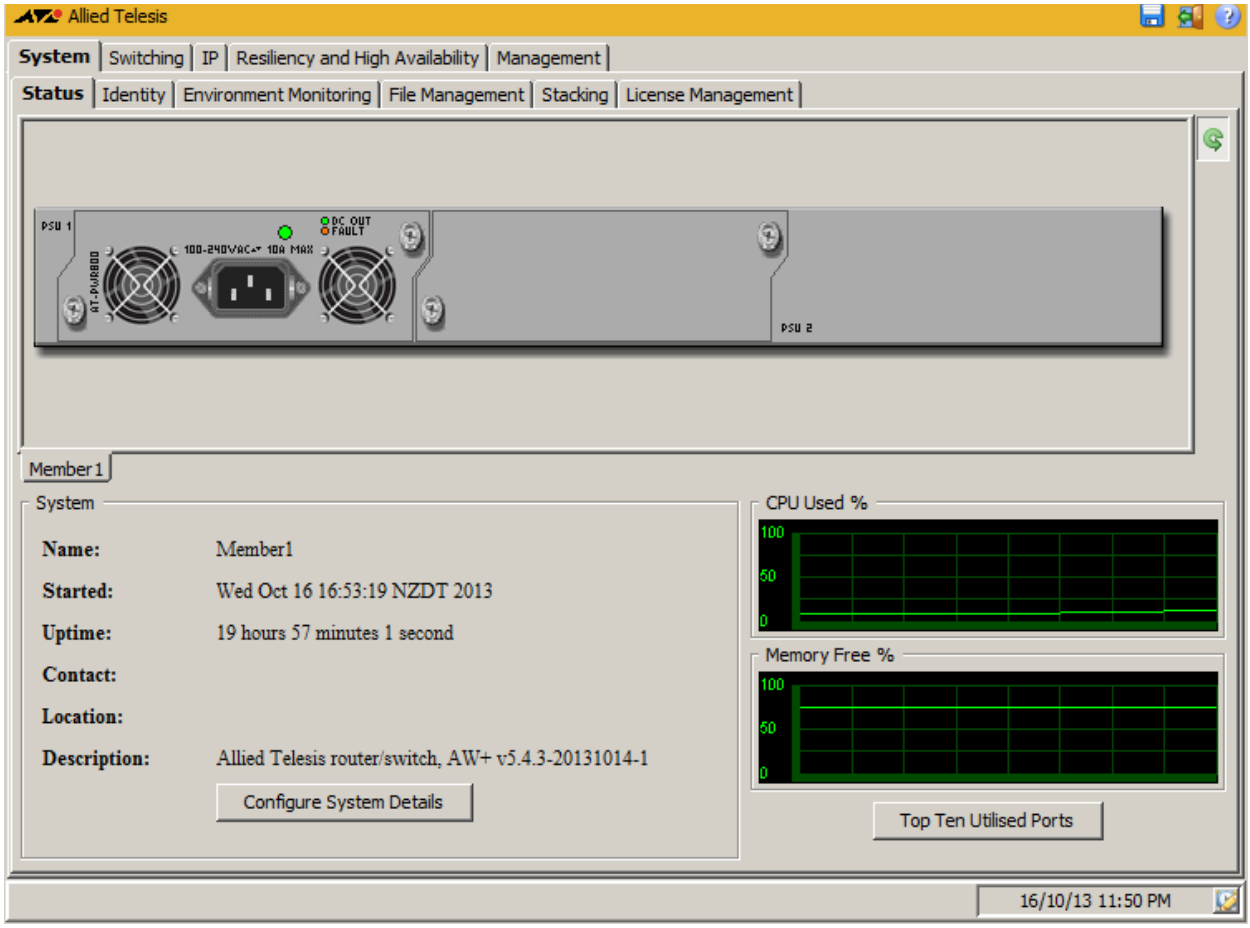
Display Label / Field	Description
View Rear Panel (icon)	Displays view of the rear panel of the chassis.
View Front Panel (icon)	Displays view of the front panel of the chassis.

Display Label / Field	Description
System / Name	Specifies the network name of the system, as set with the 'hostname' command in the CLI.
System / Started	Date and time the switch was last booted.
System / Uptime	Elapsed time since the last boot.
System / Contact	Contact details for system maintenance.
System / Location	Location of the switch
System / Description	Description of the switch, including manufacturer, model, and software version.

Description

Configuration Button / Field	Description
System Time & Date (icon)	Add or modify System Date, System Time, UTC Time Zone Offset.
Configure System Details	Add or modify System Name, System Contact, System Location.
Top Ten Utilised Ports	Displays the top ten port numbers and their utilization by percentage.

Menu Tab Example showing the **System > Status** menu tab displaying the **rear panel** of the chassis indicating on **AT-PWR-100R** PSU:



The module bay in the rear panel of the chassis has LED indicators that show the power status.

Description

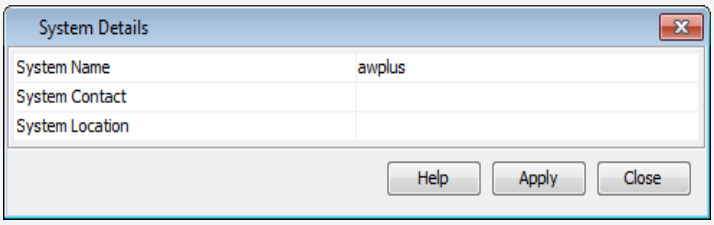
LED Indicators	Description
DC OUT	The PSU is functioning and is supplying power to the switch.
Fault	Power is not supplied to the switch from the PSU.

System > Status > System Details

The **System** > **Status** > **System Details** dialog allows you to configure basic system information.

Configuration Dialog

Example showing **System** > **Status** > **System Details** dialog:



Description

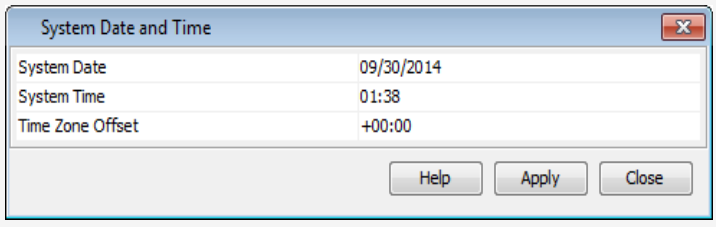
Label / Field / Button	Description
`System Name`	Enter the network name of the system.
`System Contact`	Enter the contact information for the system, from 0 to 255 characters long. Valid characters are any printable characters and spaces.
`System Location`	Enter the location of the system, from 0 to 255 characters long. Valid characters are any printable characters and spaces.

System > Status > System Date and Time

The **System** > **Status** > **System Date and Time** dialog allows you to configure the date and time for the switch.

Configuration Dialog

Example showing **System** > **Status** > **System Date and Time** dialog:



Description

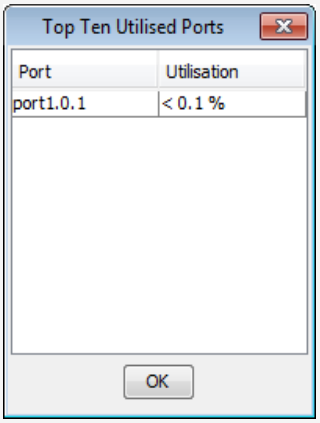
Label / Field / Button	Description
`System Date`	Enter the current system date in month, day, and year format.
`System Time`	Enter the local time for the system clock in hours and minutes.
`Time Zone Offset`	Enter the offset to the UTC (Coordinated Universal Timezone) for a local timezone in hours and minutes.

System > Status > Top Ten Utilised Ports

The **System** > **Status** > **Top Ten Utilised Port** dialog allows you to monitor port utilization on the switch.

Configuration Dialog

Example showing **System** > **Status** > **Top Ten Utilised Ports** dialog:



Description

Label / Field / Button	Description
`Port`	Displays up to ten ports that are used the most on the switch. You can sort by ascending or descending port order.
`Utilization`	Displays the utilization percentage for the port. You can sort by ascending or descending utilization percentage.

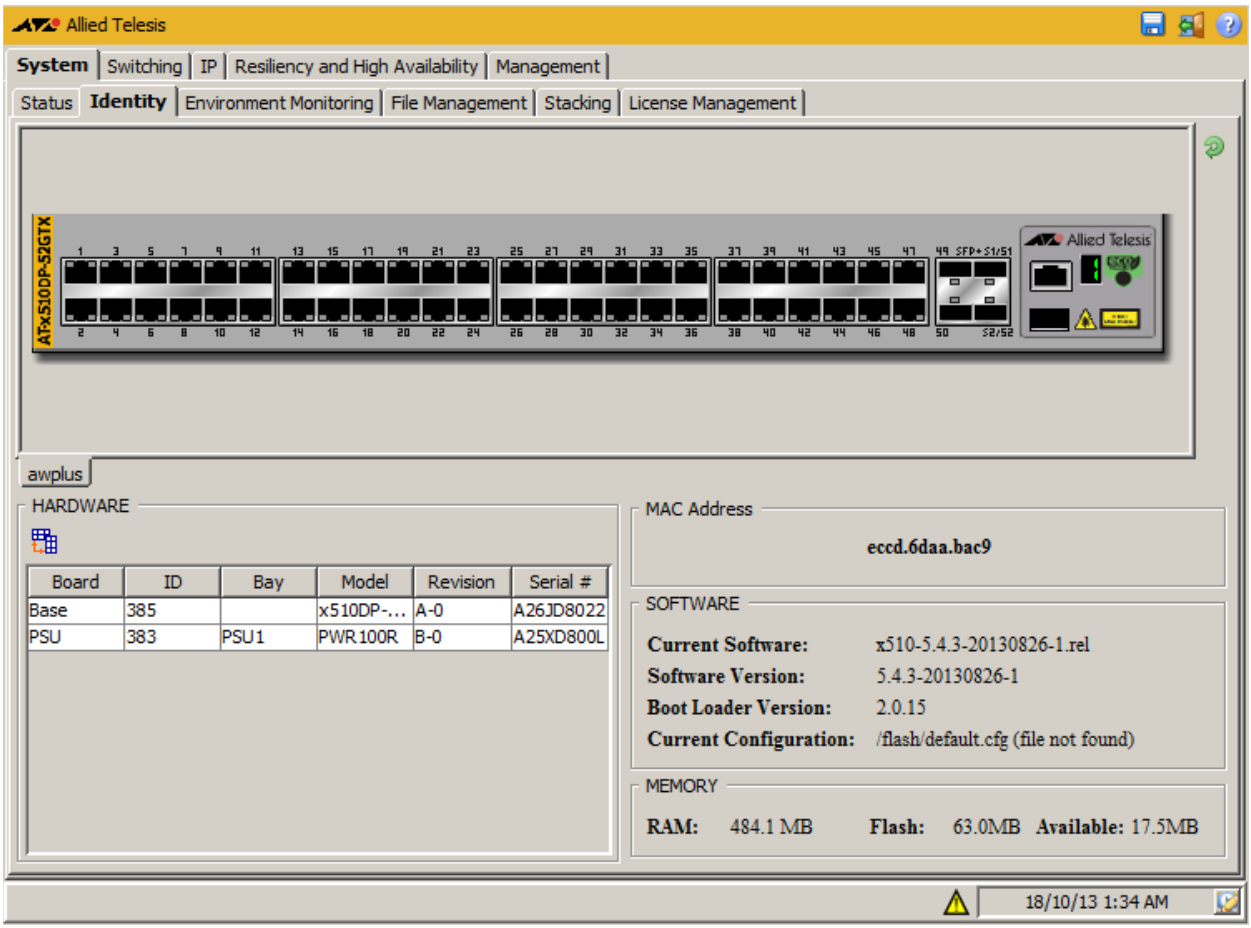
System > Identity

The **System** > **Identity** menu tab displays physical properties, software version and configuration file name.

Note

For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.
The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Example showing the **System** > **Identity** menu tab:



Description

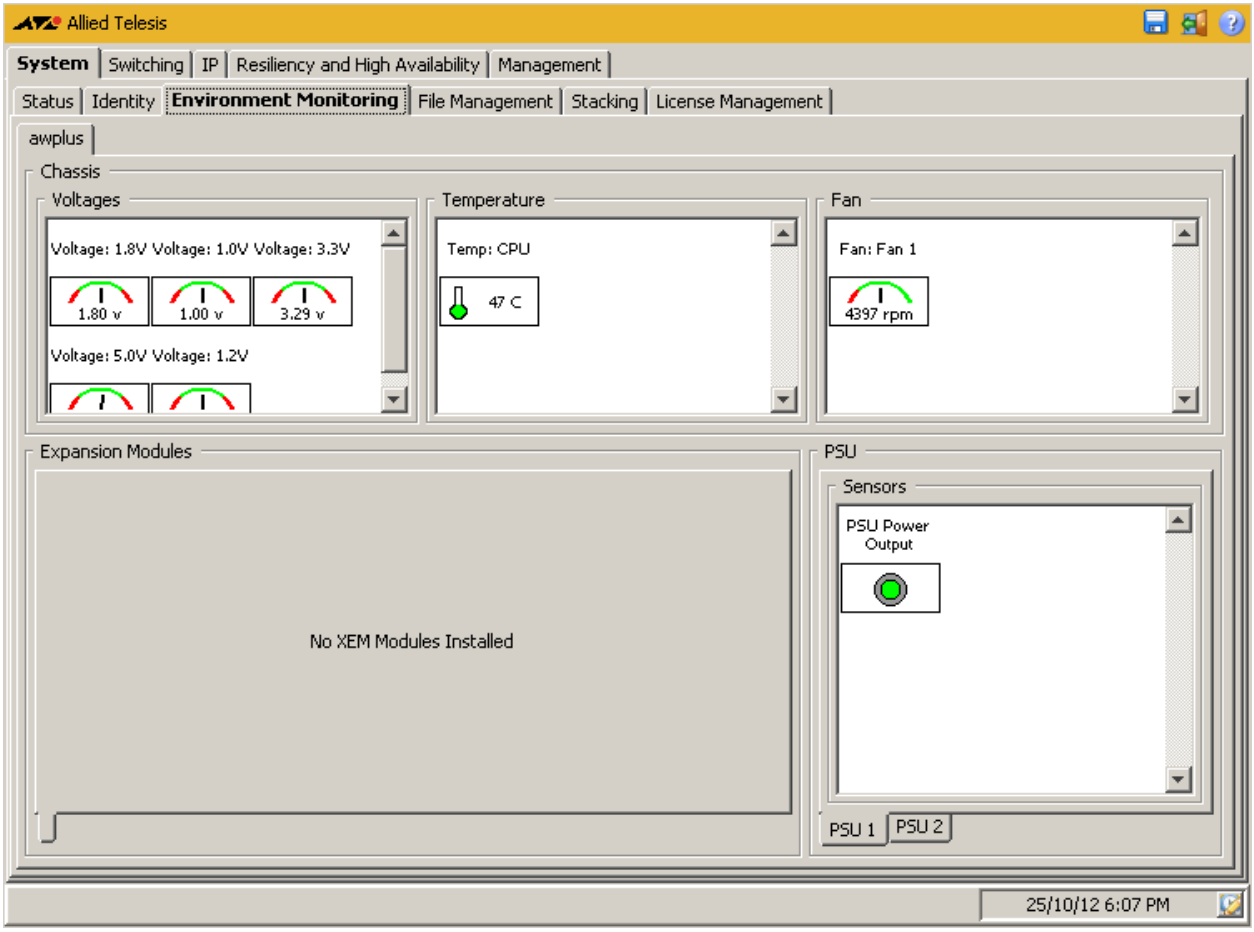
Label / Field / Button / icon	Description
HARDWARE	Displays the board, ID, bay, model, revision and serial number of the switch main board. On the x510DP-52GTX the board, ID, bay, model, revision and serial number are also shown for any installed power supply units (PSUs).
MAC Address	Displays the MAC Address of the switch in hexadecimal in the format HHHH . HHHH . HHHH.

Label / Field / Button / icon	Description(cont.)
SOFTWARE	Displays the software release file name, software version, boot loader version, and configuration file name loaded on the switch.
MEMORY	Displays the amount of installed RAM and Flash, plus the remaining Flash available on the switch.

System > Environment Monitoring

The **System** > **Environment Monitoring** menu tab allows you to display the status of the environmental properties, such as all voltages and temperatures, which the system monitors.

Menu Tab Example showing the **System** > **Environment Monitoring** menu tab:



Description

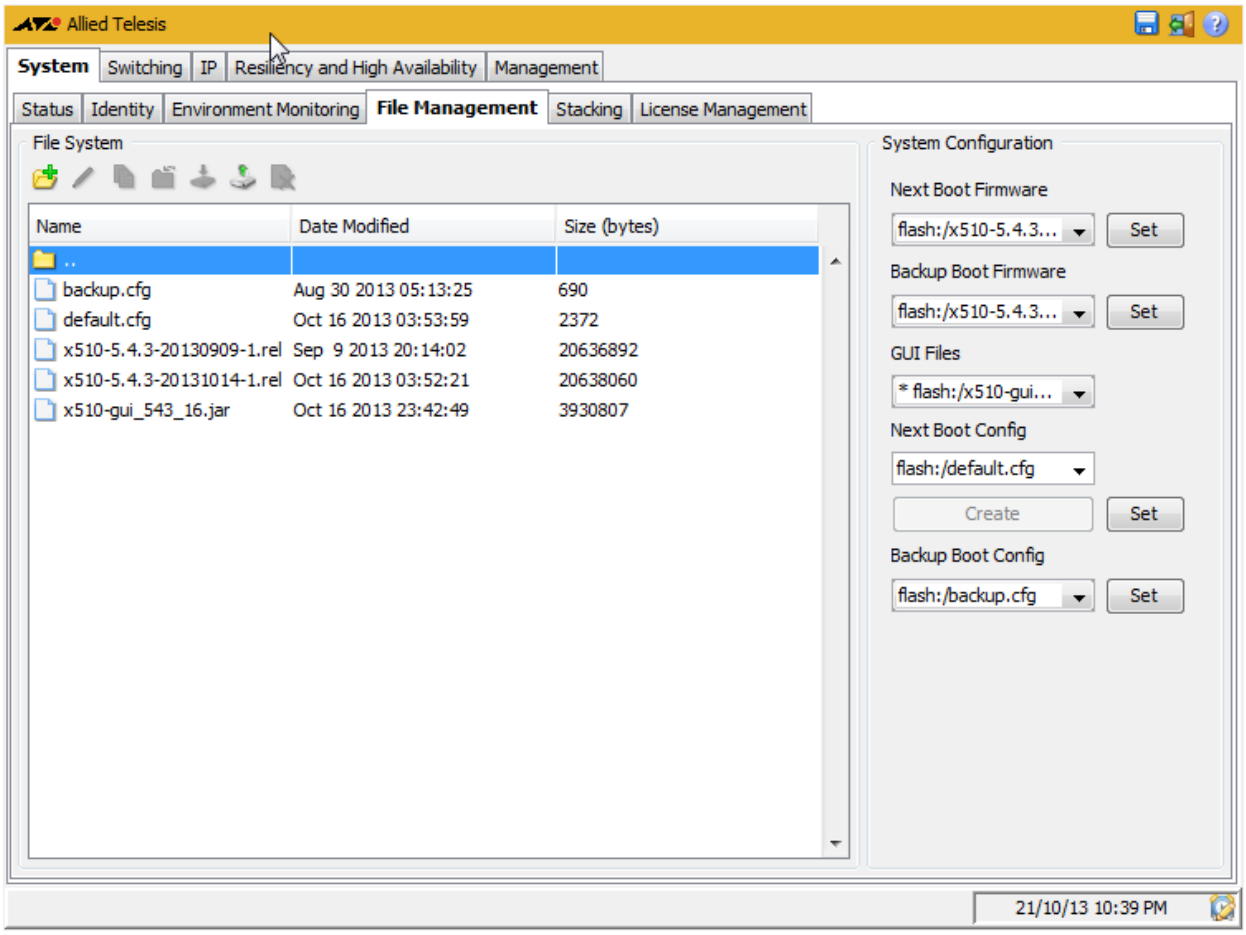
Label / Field / Button	Description
`Chassis`	Displays the operational status of chassis voltages and temperatures for the switch.
`Fan`	Displays the operational status of the switch fans.
`Expansion Modules`	This switch has no expansion modules.
`PSU`	Displays the operational status of temperatures and fans for any installed PSUs.

System > File Management

The **System > File Management** menu tab allows you to create, copy, delete, upload or download boot and backup release and configuration files to and from the switch.

You can specify fallback or backup release and configuration files in case the boot release or configuration files become corrupted, and you can also specify the boot release and configuration files to boot directly from a USB flash drive or to boot from flash.

Menu Tab Example showing the **System > File Management** menu tab:



Description: File System

Label / Field / Button	Description
`File System`	Displays file names, file dates, and file sizes of files in Flash, NVS or USB storage devices. Note that only USB storage device files that are resident on the stack master are shown. Also, only files that have a total URL length of 112 characters or less are displayed. The URL is the path to the file and is of the form `<hostname>-<stack_id>/<filesystem>:/<pathname>`, for example, `awplus-1/flash:/test.cfg`. The GUI will immediately show all file changes to the NVS and Flash filesystems, regardless of how they have been made, either via the GUI or CLI. However, the USB storage device filesystem is treated differently as it is not permanently mounted. The GUI will only update USB storage device files when the device is inserted/deleted, or when the changes are made via the GUI. They are not updated if modified via the CLI. The buttons shown below the File System label also allow you upload, download, move, copy, and delete files respectively.
`File System / Add Folder`	Select the folder you want to create a new sub-folder in then click on the Add Folder button located directly below the File System label.
`File System / Rename File`	Select the file you want to rename then click on the Rename File or Folder button located directly below the File System label.
`File System / Copy File`	Select the file you want to copy then click on the Copy File button located directly below the File System label. Choose the Destination Folder from the drop down list in the Copy File dialog then select OK to copy the file to the chosen destination.
`File System / Move File`	Select the file you want to move then click on the Move File button located directly below the File System label. Choose the Destination Folder from the drop down list in the Move File dialog then select OK to move the file to the chosen destination.
`File System / Download File`	Select the file you want to download then click on the Download File button located directly below the File System label.
`File System / Upload File`	Click on the Upload File button located directly below the File System label then select the file you want to upload.
`File System / Delete File`	Select the file or folder you want to delete then click on the Delete File button located directly below the File System label.

Description: System Configuration

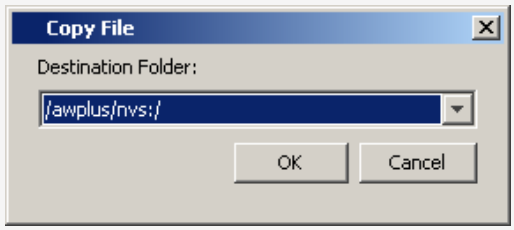
Label / Field / Button	Description
`System Configuration`	Configures running and backup software, GUI software, and configuration files in Flash or card (or storage device) memory available on the switch.
`System Configuration / Next Boot Firmware`	Choose the Next Boot Firmware `.rel` file and path from the drop down list then click Set to make this file the firmware that starts after reboot. You can set a Next Boot Firmware `.rel` file to boot directly from a USB storage device.
`System Configuration / Backup Boot Firmware`	Choose the Backup Boot Firmware `.rel` file and path from the drop down list then click Set to boot from this file at reboot. A Backup Boot Firmware .rel file is used instead of the Next Boot Firmware `.rel` file if the Next Boot Firmware `.rel` file is corrupted.
`System Configuration / GUI Files`	Displays the GUI file name and file location on the switch and indicates the currently running GUI file with a prefixed asterisk (e.g. `* flash:/x510-gui_543_10.jar`). Note that you cannot set the GUI version from within the GUI itself. See the GUI installation instructions in *Appendix C: GUI Reference* of the current *AlliedWare Plus Software Reference* to install GUI files. The latest version of the GUI .jar file loaded is run by the switch automatically.
`System Configuration / Next Boot Config`	Choose the Next Boot Config `.cfg` file and path from the drop down list then click Set to make this file the Config `.cfg` file that the switch uses at reboot. You can set a Next Boot Config `.cfg` file to load directly from a USB storage device.
`System Configuration / Backup Boot Config`	Choose the Backup Boot Config Files .cfg file and path from the drop down list then click Set to make boot from this file at reboot. A Backup Boot Config `.cfg` file is used instead of the Next Boot config `.cfg` file if the Next Boot Config `.cfg` file is corrupted.

System > File Management > Copy File

The **System** > **File Management** > **Copy File** dialog allows you to copy files (e.g. release and configuration files). Select the 'Copy File' button below the 'File System' label to access this dialog.

Configuration Dialog

Example showing **System** > **File Management** > **Copy File** dialog:



Description

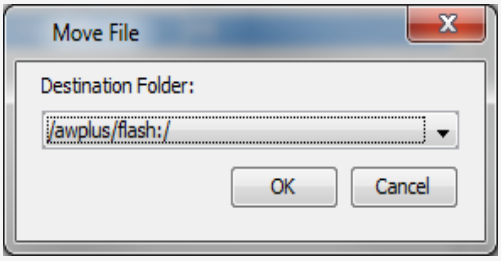
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to copy the selected file to. A copying progress bar displays after clicking the OK button.

System > File Management > Move File

The **System** > **File Management** > **Move File** dialog allows you to move files (e.g. release and configuration files). Select the 'Move File' button below the 'File System' label to access this dialog.

Configuration Dialog

Example showing **System** > **File Management** > **Move File** dialog:



Description

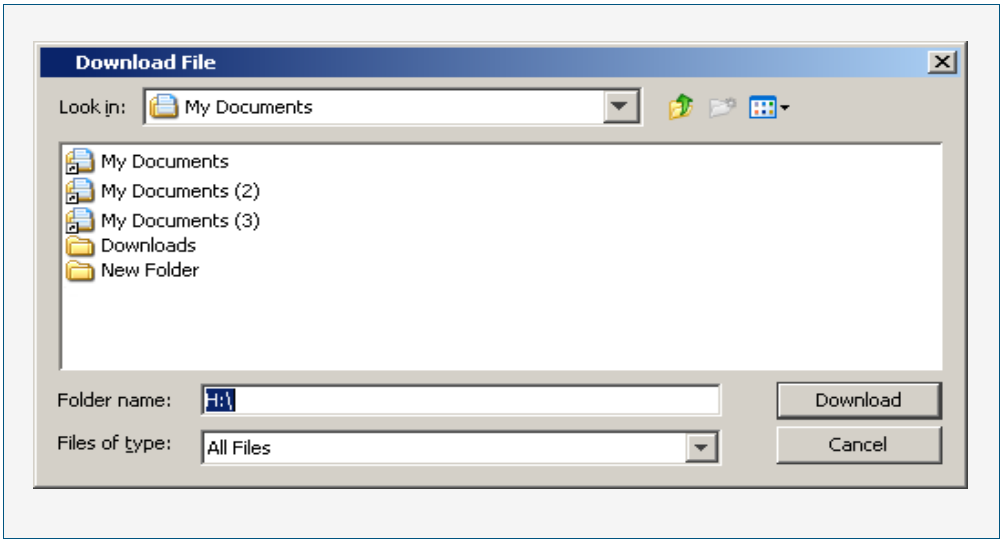
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to move the selected file to. A moving progress bar displays after clicking the OK button.

System > File Management > Download File

The **System > File Management > Download File** dialog allows you to download files (e.g. release and configuration files) from the switch to a client device. Select the 'Download File' button below the 'File System' label to access this dialog.

Configuration Dialog

Example showing **System > File Management > Download File** dialog:



Description

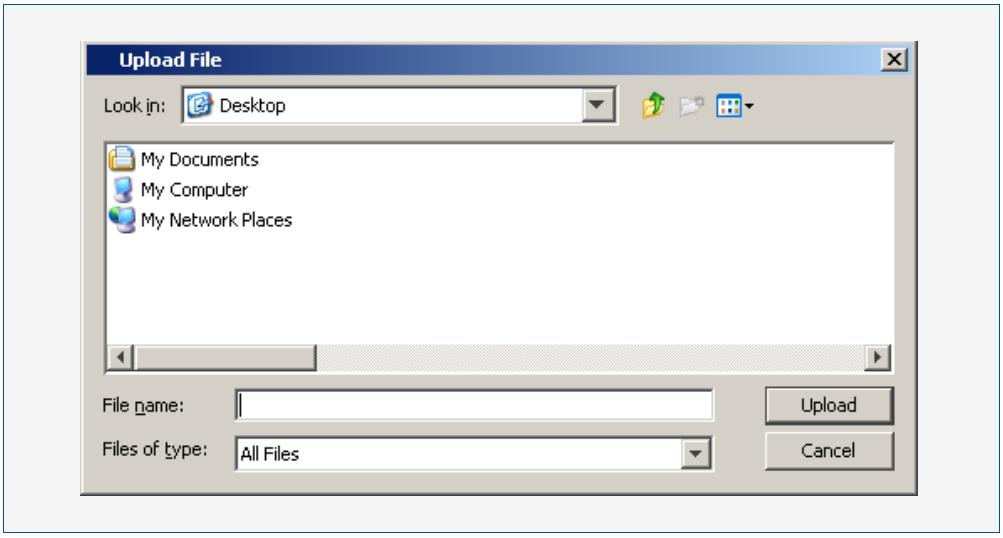
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to download the selected file to. A downloading progress bar displays after clicking the Download button.

System > File Management > Upload File

The **System** > **File Management** > **Upload File** dialog allows you to upload files (e.g. release and configuration files) from a client device to the switch. Select the 'Upload File' button below the 'File System' label to access this dialog.

Configuration Dialog

Example showing **System** > **File Management** > **Upload File** dialog:



Description

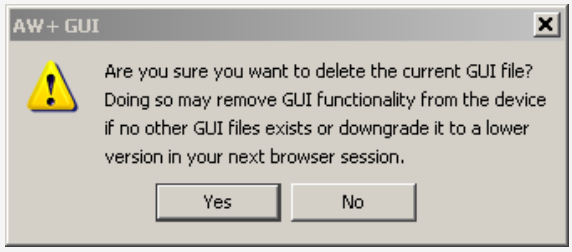
Label / Field / Button	Description
`Destination Folder`	Select the destination folder to upload the selected file from. An uploading progress bar displays after clicking the Upload button.

System > File Management > Delete File

The **System** > **File Management** > **Delete File** dialog allows you to delete files (e.g. release and configuration files). Select the 'Delete File' button below the 'File System' label to access this dialog.

Configuration Dialog

Example showing **System** > **File Management** > **Delete File** dialog:



Description

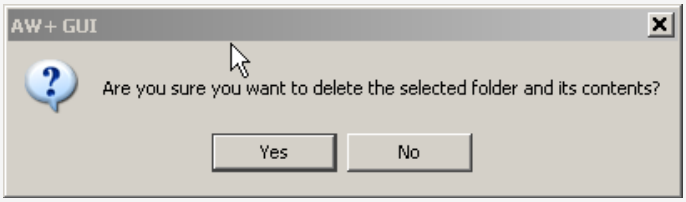
Label / Field / Button	Description
Yes	Confirm selected file deletion operation.
No	Cancel selected file deletion operation.

System > File Management > Delete Folder

The **System** > **File Management** > **Delete Folder** dialog allows you to delete folders in the flash or USB storage device file system containing files (e.g. release and configuration files). Select the 'Delete Folder' button below the 'File System' label to access this dialog.

Configuration Dialog

Example showing **System** > **File Management** > **Delete Folder** dialog:



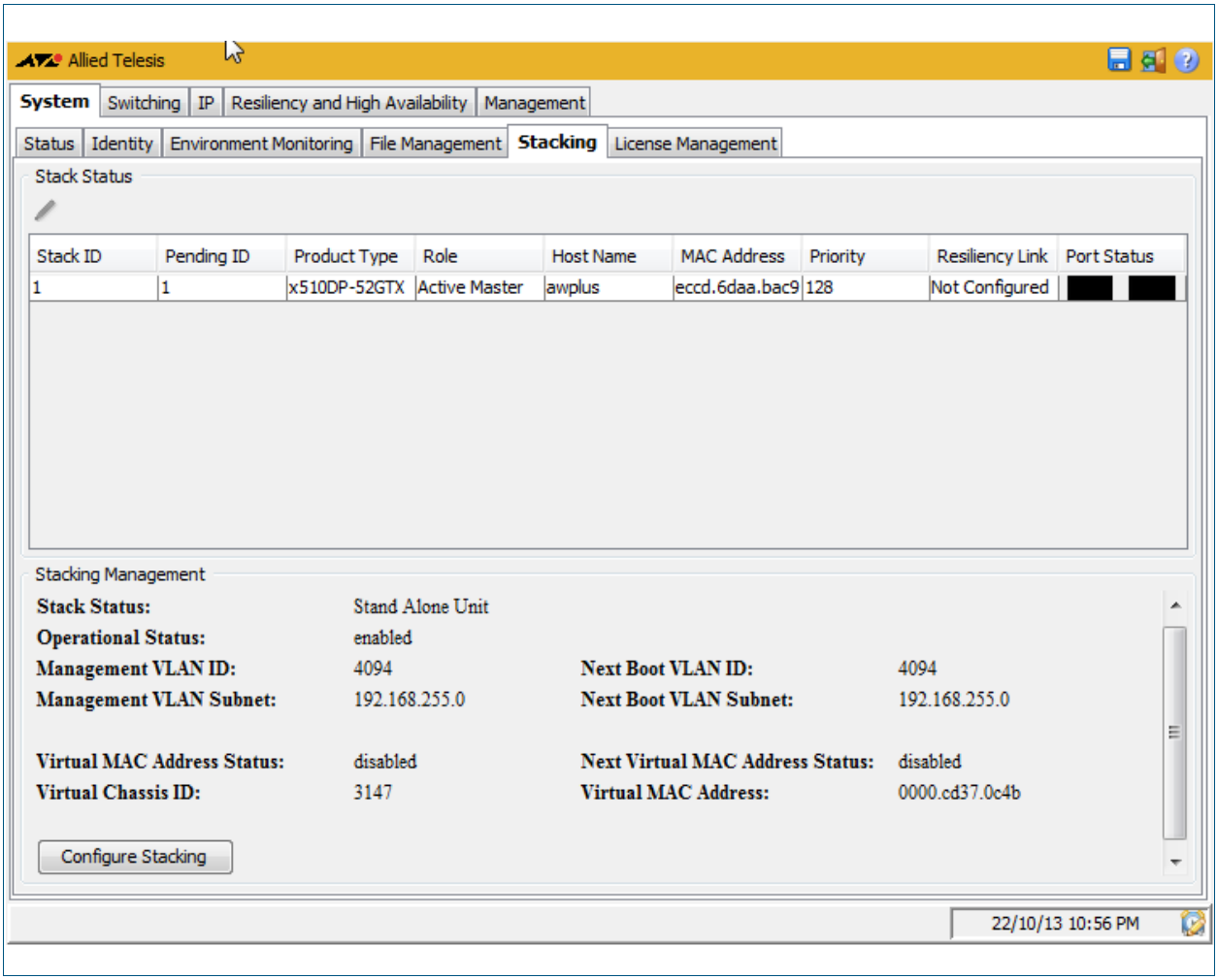
Description

Label / Field / Button	Description
Yes	Confirm selected folder deletion operation.
No	Cancel selected folder deletion operation.

System > Stacking

The **System** > **Stacking** menu tab allows you to display and monitor a summary of the identity and status of stack members, plus you can also configure the VLAN ID and IP subnets used for internal VCStack communication.

Menu Tab Example showing the **System** > **Stacking**:



Description: Stacking Management

Label / Field / Button	Description
`Stacking Management / Stack Status`	The stack's overall status. Note that a warning is issued if the stack is not connected in a standard ring topology.
`Stacking Management / Operational Status`	The operational status of the stack. Can be: enabled (1) disabled (2)
`Stacking Management / Management VLAN ID`	The VLAN ID currently used for stack management. The default stack management VLAN ID is 4094.
`Stacking Management / Next Boot VLAN ID`	The VCS management VLAN ID to be assigned after the next reboot.

Label / Field / Button(cont.)	Description(cont.)
Stacking Management / Management VLAN Subnet	The VLAN subnet currently used for stack management.
Stacking Management / Next Boot VLAN Subnet	The stacking management VLAN subnet address after rebooting.
Stacking Management / Virtual MAC Address Status	Indicates whether the virtual MAC address is enabled or disabled.
Stacking Management / Next Virtual MAC Address Status	Indicates whether the next virtual MAC address is enabled or disabled.
Stacking Management / Virtual Chassis ID	Displays the current virtual chassis ID.
Stacking Management / Virtual MAC Address	Displays the virtual MAC address used by the stack.
Configure Stacking	Configures the VCS management VLAN ID, the subnet address of the VCS management VLAN, and the Virtual MAC Address Status.

Description: Stack Status

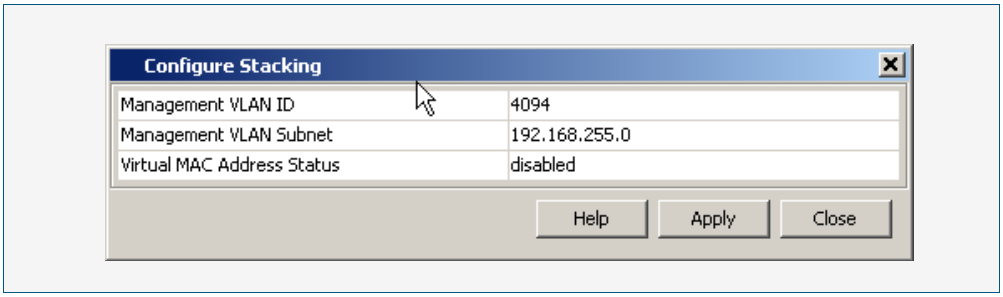
Label / Field / Button	Description
`Stack Status / Stack ID`	The Stack member ID.
`Stack Status / Pending ID`	The Stack member ID to be assigned to the device after the next reboot.
`Stack Status / Product Type`	The Stack member product type; for example, SwitchBlade x908.
`Stack Status / Role`	Stack member's role in the stack (either Active Master or Backup Member).
`Stack Status / Host Name`	The host name of the Stack member.
`Stack Status / MAC Address`	Stack member's hardware MAC address. Note that frames from devices within a stacked virtual chassis will carry the source address of the stack master.
`Stack Status / Priority`	The priority for election of stack master (0 to 255). The lowest number has the highest priority. Note that where stack members have the same priority setting, the switch with the lowest MAC address will become the stack master.
`Stack Status / Resiliency Link`	Status of the stack members resiliency link. Can be one of: `Configured`, `Successful`, `Failed`, `Not Configured`.
`Stack Status / Port Status`	When the rectangle is colored GREEN, it means that the port has a learned neighbor connected. Note that the number in the rectangle indicates the stack ID of the learned neighbor connected to the port. When the rectangle is colored BLACK then the port status is down. When the rectangle is colored RED then Operational Status for the port has been set to disabled.

System > Stacking > Configure Stacking

The **System** > **Stacking** > **Configure Stacking** dialog allows you to configure the VLAN ID and IP subnets used for internal VCStack communication.

Configuration Dialog

Example showing **System** > **Stacking** > **Configure Stacking** dialog:



Description

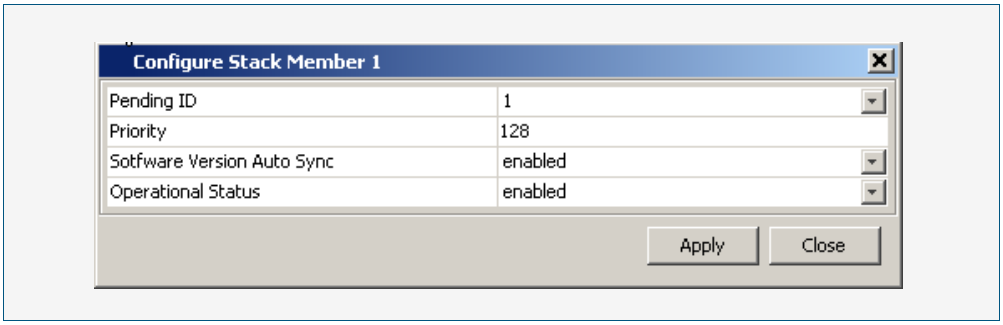
Label / Field / Button	Description
`Management VLAN ID`	Enter the VLAN ID for stack management. The default stack management VLAN ID is 4094.
`Management VLAN subnet`	Enter the subnet address of the VCS management VLAN.
`Virtual MAC Address Status`	Select the status for the Virtual MAC Addressing feature, which can be set to `disabled` or `enabled` from the drop down list. You can only change this field if the `Operational Status` field is enabled, otherwise this field is disabled. Note also that when running VCStack configurations via the GUI, you must enable Virtual MAC Addressing

System > Stacking > Configure Stack Member

The **System** > **Stacking** > **Configure Stack Member** dialog allows you to configure the Pending ID, Priority, Software Version Auto Synchronization and Operational Status used for internal VCStack communication.

Configuration Dialog

Example showing **System** > **Stacking** > **Configure Stack Member** dialog:



Description

Label / Field / Button	Description
`Pending ID`	Enter the Pending ID for the stack member. You can only change this field if the `Operational Status` field is enabled, otherwise this field is disabled.
`Priority`	Enter the Priority for the stack member.
`Software Version Auto Sync`	Select the enabled or disabled options to enable or disable the Software Version Auto Synchronization feature for the stack member. You can only change this field if the `Operational Status` field is enabled, otherwise this field is disabled.
`Operational Status`	Select the enabled or disabled options to enable or disable the stack member.

System > License Management

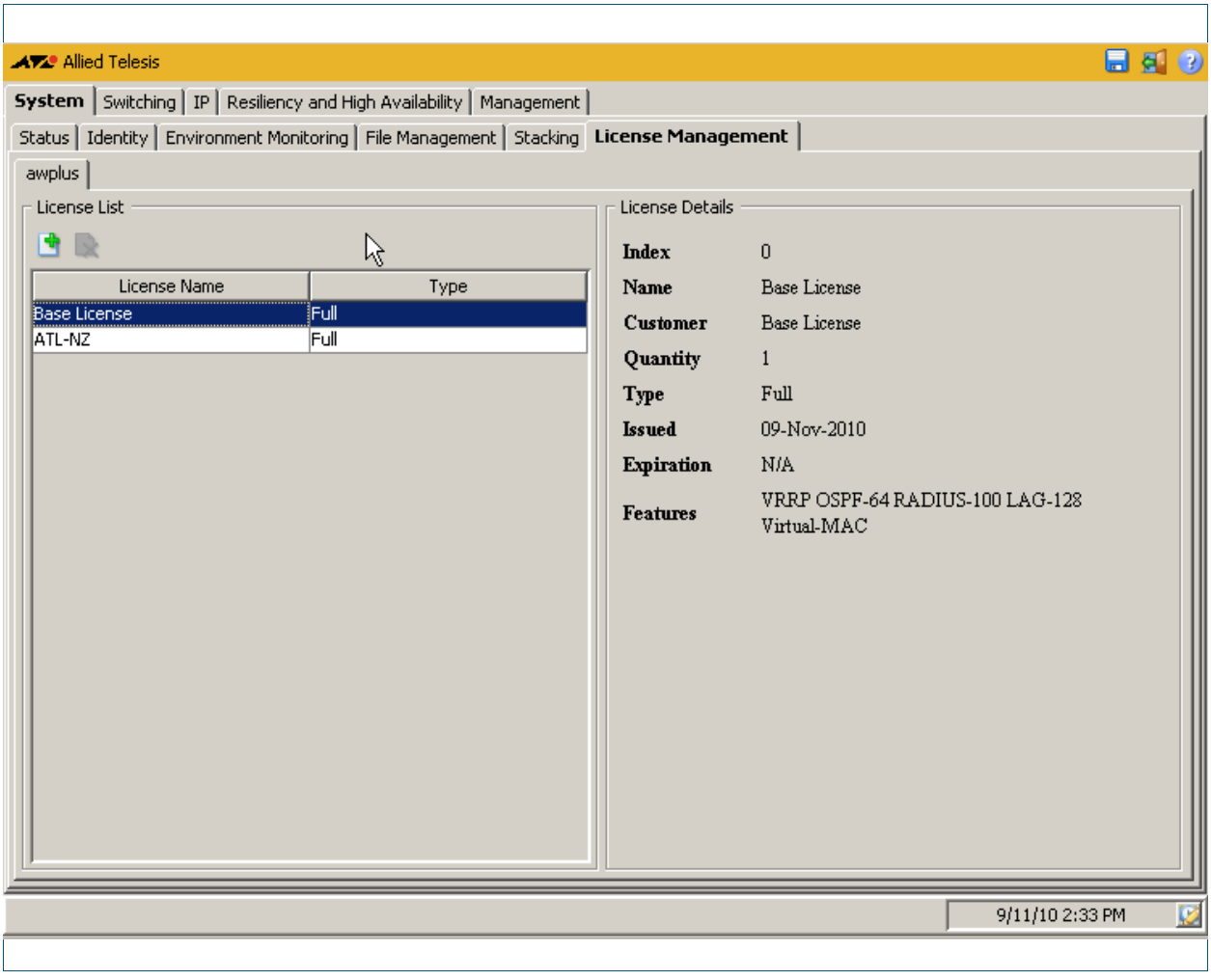
The **System** > **License Management** menu tab allows you to view, add and delete feature licenses.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.

- Selecting the + icon allows you to add a feature license.
- Selecting the x icon allows you to delete a feature license.

Note If a license is added to, or deleted from, a stack member then the same action must be taken on all other stack members. Otherwise incompatible licensing will occur and affected devices will not rejoin the stack following a reboot.

Menu Tab Example showing the **System** > **License Management** menu tab:



Description

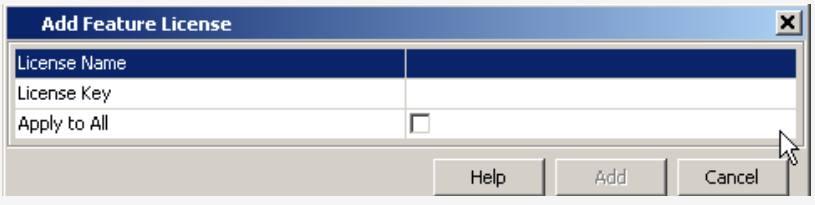
Label / Field / Button	Description
`License List / License Name`	Name of the license bundle.
`License List / Type`	The type of license activated on the switch: full or temporary.
`License Details / Index`	Index identifying entry.
`License Details / Name`	Name of the license bundle.
`License Details / Customer`	Customer name.
`License Details / Quantity`	Quantity of licenses included in the feature key.
`License Details / Type`	Full or temporary license types.
`License Details / Issued`	Date the key was generated.
`License Details / Expiration`	Expiry date for a temporary license.
`License Details / Features`	List of features enabled by the license.

System > License Management > Add Feature License

The **System > License Management > Add Feature License** dialog allows you add feature licenses by specifying the license name and the license key. You can add a license for all the stack members or for a single stack member.

Configuration Dialog

Example showing **System > License Management > Add Feature License** dialog:



Description

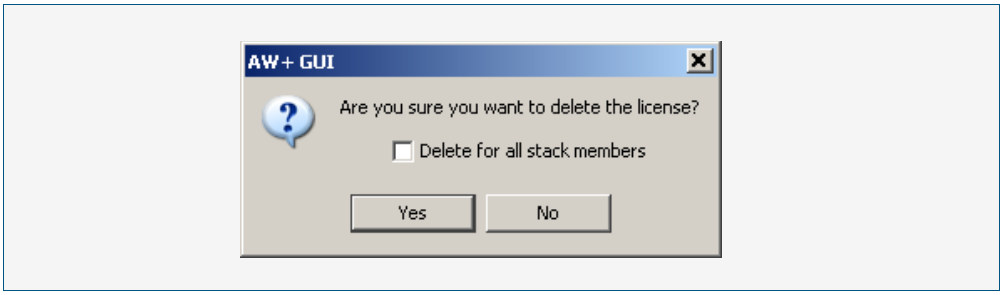
Label / Field / Button	Description
`License Name`	Enter the license name of the software feature.
`License Key`	Enter the encrypted license key to enable this software feature.
`Apply to All`	Select the checkbox to apply the license to all the stack members.

System > License Management > Delete Feature License

The **System > License Management > Delete Feature License** dialog allows you delete feature licenses by specifying the license name and the license key. You can delete a license for all the stack members or for a single stack member.

Configuration Dialog

Example showing **System > License Management > Delete Feature License** dialog:



Description

Label / Field / Button	Description
`Delete for all stack members`	Select the checkbox to delete the license for all the stack members.

Switching > Ports

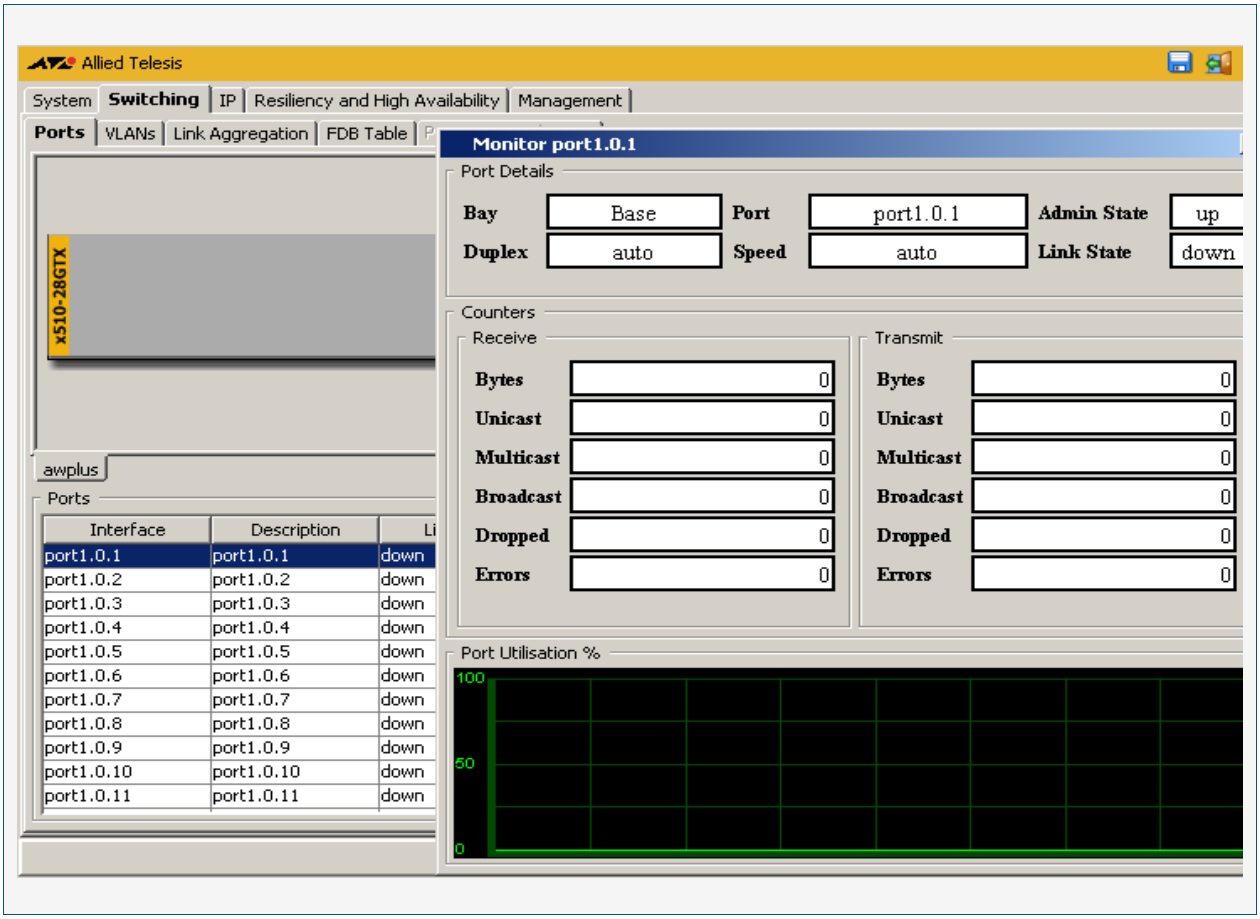
The **Switching** > **Ports** menu tab allows you to view, and configure Layer 1 properties:

- Right-clicking a port allows you to select monitoring or configuration dialogs for the selected port.
- The monitoring dialog displays port status, statistics and a brief utilization history.
- The configuration dialog allows you to configure Administrative State, Auto Negotiation, Speed and Duplex settings for the port.

Note Speed and Duplex settings can only be changed if Auto Negotiation is disabled.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.
The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Example showing the **Switching > Ports** menu tab and **Monitor**:



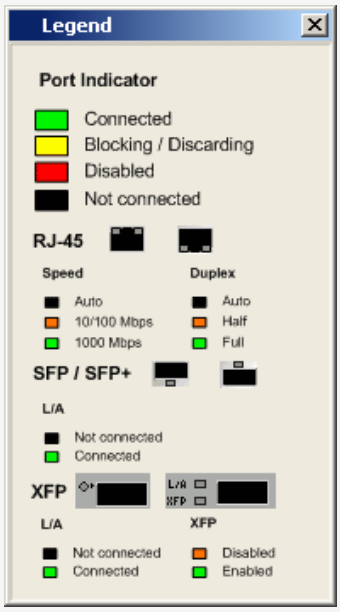
Description

Label / Field / Button	Description
`Ports`	Displays port number, description of the port, link and administrative status, duplex mode, speed and uptime (in milliseconds) for the selected port.

Legend

Example showing S**witching > Ports > Legend**:

To select the Legend panel click the View Legend icon (white L within a blue circle) on the top right of the Switching - Ports screen.

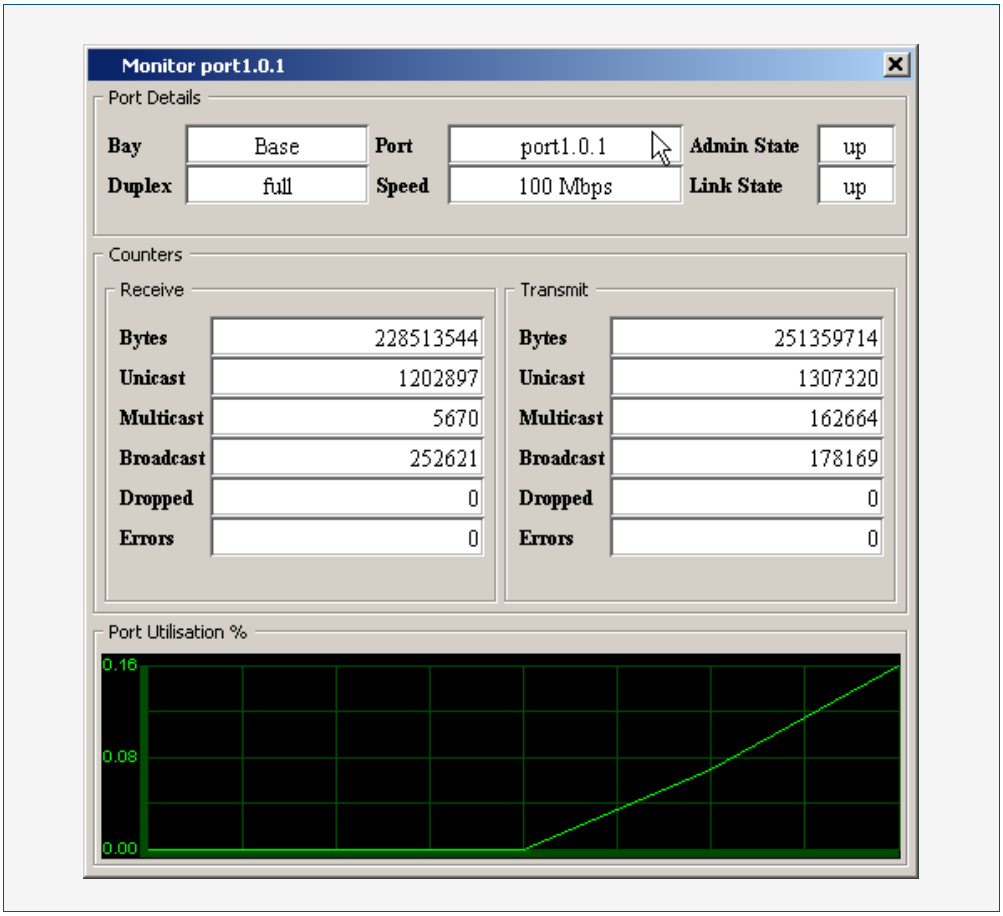


Switching > Ports > Monitor Port

The **Switching** > **Ports** > **Monitor Port** dialog allows you to monitor port counters.

Configuration Dialog

Example showing the **Switching** > **Ports** > **Monitor Port** dialog:



Description

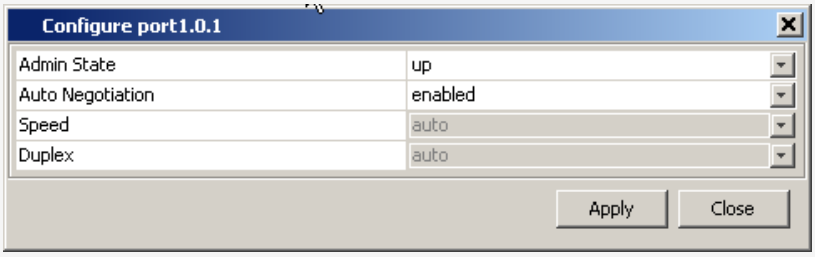
Label / Field / Button	Description
`Port Details`	Monitors the bay, port, duplex, speed, administrative state and link states for the selected port.
`Counters`	Monitors the counters for bytes received/transmitted, unicast packets received/transmitted, multicast packets received/transmitted, broadcast packets received/transmitted, dropped packets received/transmitted, and errors received/transmitted for the selected port.
`Port Utilisation%`	Monitors and graphs the usage percentage for the selected port.

Switching > Ports > Configure Port

The **Switching** > **Ports** > **Configure Port** dialog allows you to configure Administrative State, Auto Negotiation, Speed and Duplex settings for the selected port.

Configuration Dialog

Example showing the **Switching** > **Ports** > **Configure Port** dialog:



Description

Label / Field / Button	Description
`Admin State`	Select `up` or `down` from the drop down list in this dialog to specify the administrative state for the selected port.
`Auto Negotiation`	Select `disabled` or `enabled` from the drop-down list in this dialog to specify auto negotiation for the selected port. Note that selecting `enabled` to enable `Auto Negotiation` will disable Speed and Duplex options, indicated by greyed out options.
`Speed`	Select `10 Mbps`, `100 Mbps`, `1000 Mbps`, `10 Gbps`, or `auto` from the drop down list in this dialog to specify the speed setting for the selected port. Note that the options for speed settings are only available if `Auto Negotiation` has been `disabled` for the selected port.
`Duplex`	Select `full`, `half`, or `auto` from the drop down list in this dialog to specify the duplex setting for the selected port. Note that the options for duplex settings are only available if `Auto Negotiation` has been `disabled` for the selected port.

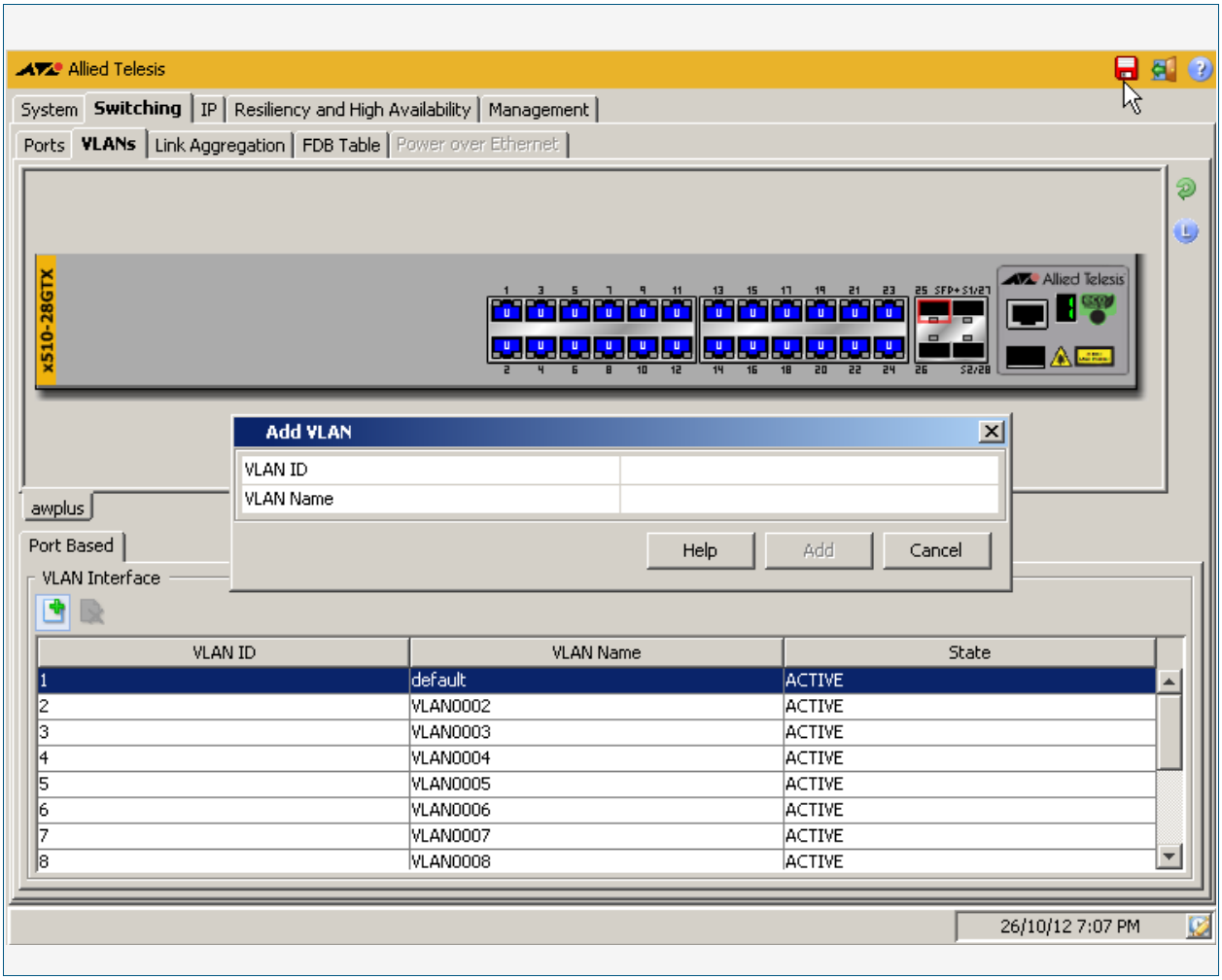
Switching > VLANs

The **Switching** > **VLANs** menu tab allows you to view and configure Layer 2 properties:

- Right-clicking a port allows you to select a VLAN to be tagged or untagged for the port, or to remove a port from the VLAN.
- Define VLANs before assigning VLANs to ports on the front panel of the switch.
- Selecting the **+** icon (under the VLAN Interface label on the VLAN tab below the front panel illustration) allows you to add a VLAN by specifying the VLAN ID and VLAN Name.
- Selecting the **x** icon (under the VLAN Interface label below the front panel illustration) allows you to delete a VLAN (except for the default VLAN 1 that is assigned to all ports).

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.
The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

Menu Tab Example showing the **Switching** > **VLANs** menu tab:



Description

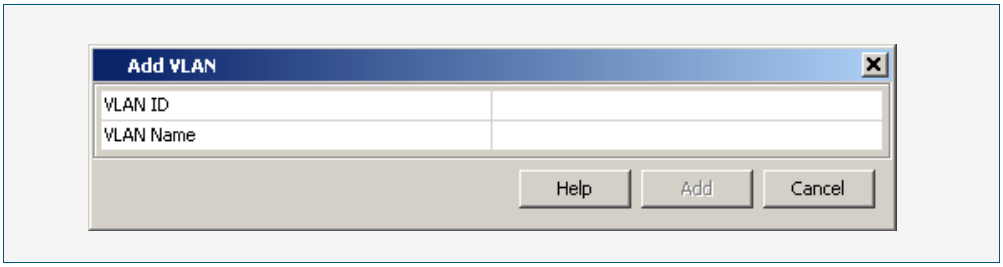
Label / Field / Button	Description
`Port Based / VLAN ID`	The VID of the VLAN that is enabled or disabled in the range 1-4094.
`Port Based / VLAN Name`	The ASCII name of the VLAN with a maximum length of 32 characters.
`Port Based / State`	The state of the VLAN, either enabled ('**ACTIVE**' displayed) or disabled ('**INACTIVE**' displayed).

Switching > VLANs > Add VLAN

The **Switching** > **VLANs** > **Add VLAN** dialog allows you add a VLAN by specifying the VLAN ID and VLAN Name.

Configuration Dialog

Example showing the **Switching** > **VLANs** > **Add VLAN** dialog:



Description

Label / Field / Button	Description
`VLAN ID`	Enter the VID of the VLAN that is enabled or disabled in the range <1-4094>.
`VLAN Name`	Enter the ASCII name of the VLAN with a maximum length of 32 characters.

Switching > Link Aggregation

The **Switching** > **Link Aggregation** menu tab allows you to view, and configure Layer 2 properties:

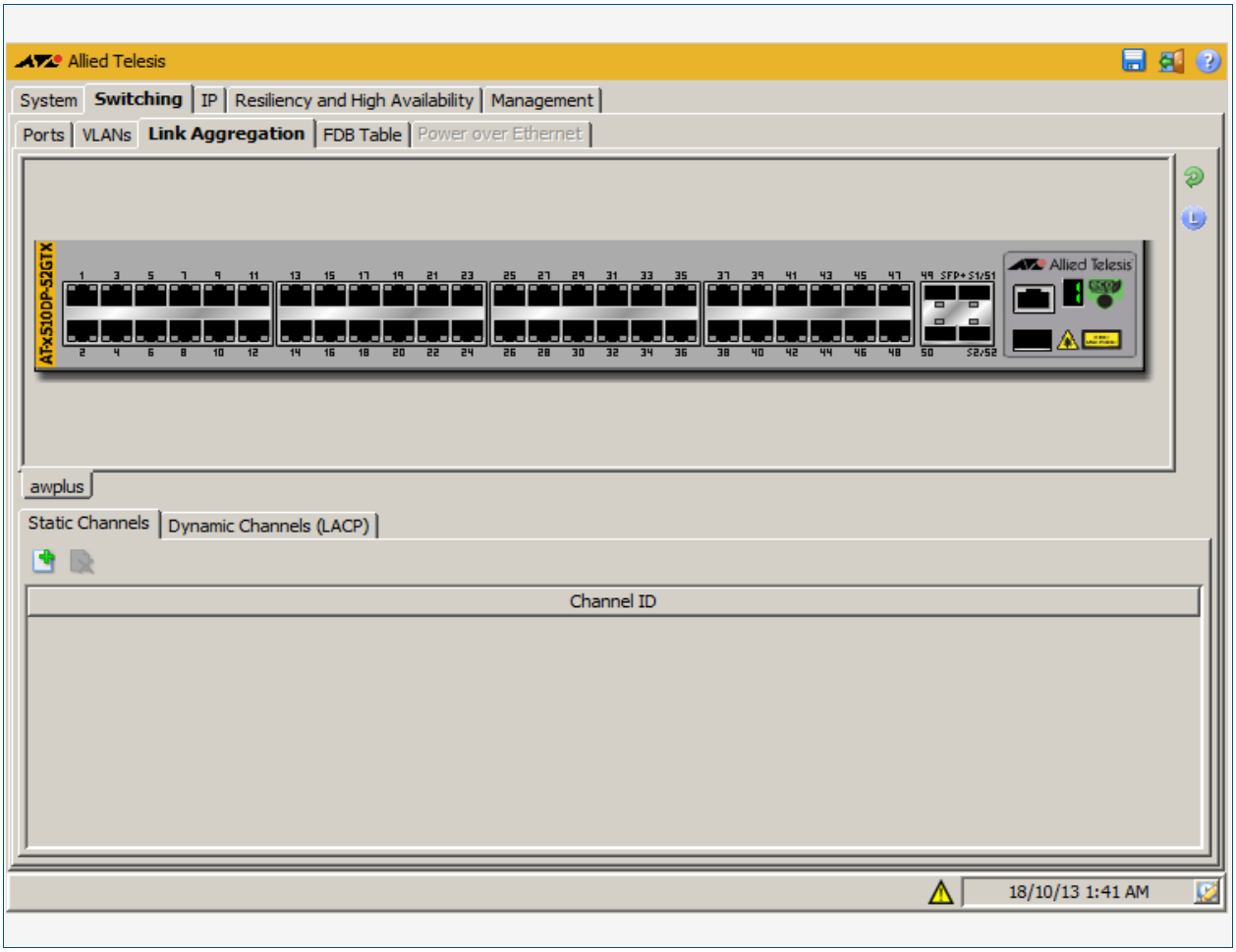
- Right-clicking a port allows you to select, assign or remove the port to a Static Channel or a Dynamic Channel (LACP - Link Aggregation Control Protocol) group.
- Define Static Channel or Dynamic Channel (LACP) groups before assigning them to ports on the front panel of the switch.
- Selecting the **+** icon (located below the front panel illustration of your switch) allows you to add a Static Channel or Dynamic Channel (LACP) group by specifying the Channel ID.

Note Up to 96 Static Channel groups and up to 32 Dynamic Channel (LACP) groups can be defined on a switch.

- Selecting the **x** icon (located below the front panel illustration of your switch) allows you to delete a Static Channel or Dynamic Channel (LACP) group.

Note For systems equipped and configured using VCStack there is a separate tab for each stack member with the system name displayed on each tab.
The last two SFP+ port LEDs are lit when VCStack is enabled on the switch.

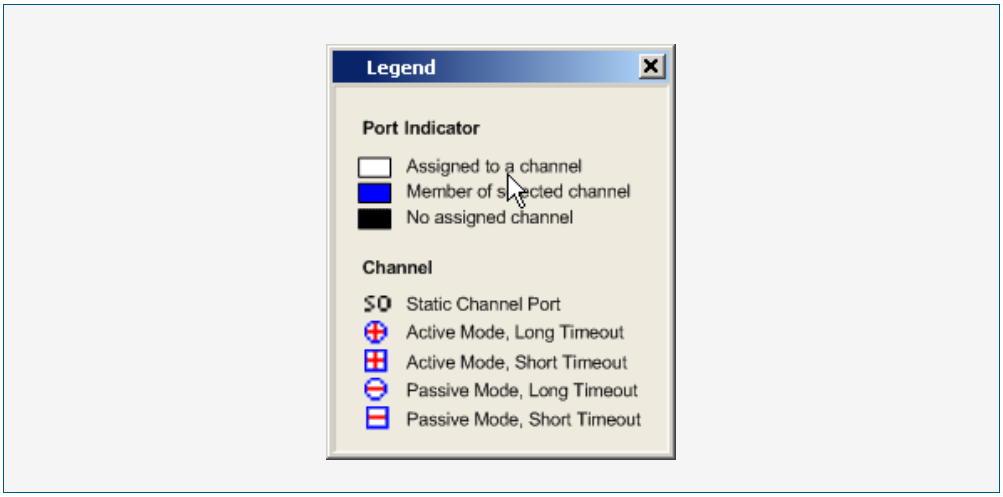
Menu Tab Example showing the **Switching > Link Aggregation** menu tab:



Description

Label / Field / Button	Description
`Static Channels`	Display or specify a static channel group number for an aggregated link. Up to 96 static channel groups can be created on the switch.
`Dynamic Channels (LACP)`	Display or specify a dynamic channel group number for an LACP link. Up to 32 dynamic channel groups can be created on the switch.

Legend Example showing **Switching > Link Aggregation > Legend**:

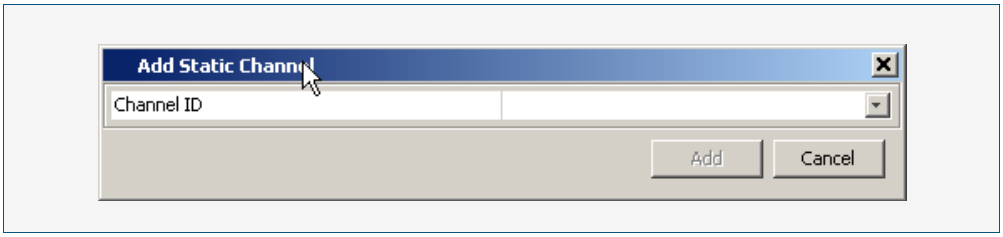


Switching > Link Aggregation > Add Static Channel

The **Switching** > **Link Aggregation** > **Add Static Channel** dialog allows you to assign the selected port to a Static Channel group.

Configuration Dialog

Example showing the **Switching** > **Link Aggregation** > **Add Static Channel** dialog:



Description

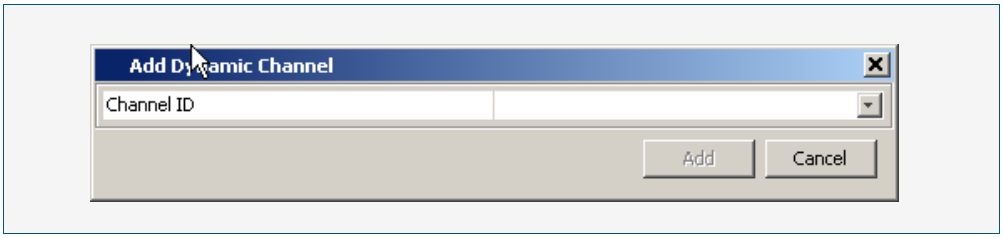
Label / Field / Button	Description
`Channel ID`	Specify a static channel group number for an interface. Up to 96 static channel groups can be created on the switch.

Switching > Link Aggregation > Add Dynamic Channel

The **Switching** > **Link Aggregation** > **Add Dynamic Channel** dialog allows you to assign the selected port to a Dynamic Channel (LACP) group.

Configuration Dialog

Example showing **Switching** > **Link Aggregation** > **Add Dynamic Channel** dialog:



Description

Label / Field / Button	Description
`Channel ID`	Specify a dynamic (LACP) channel group number for an interface. Up to 32 dynamic (LACP) channel groups can be created on the switch.

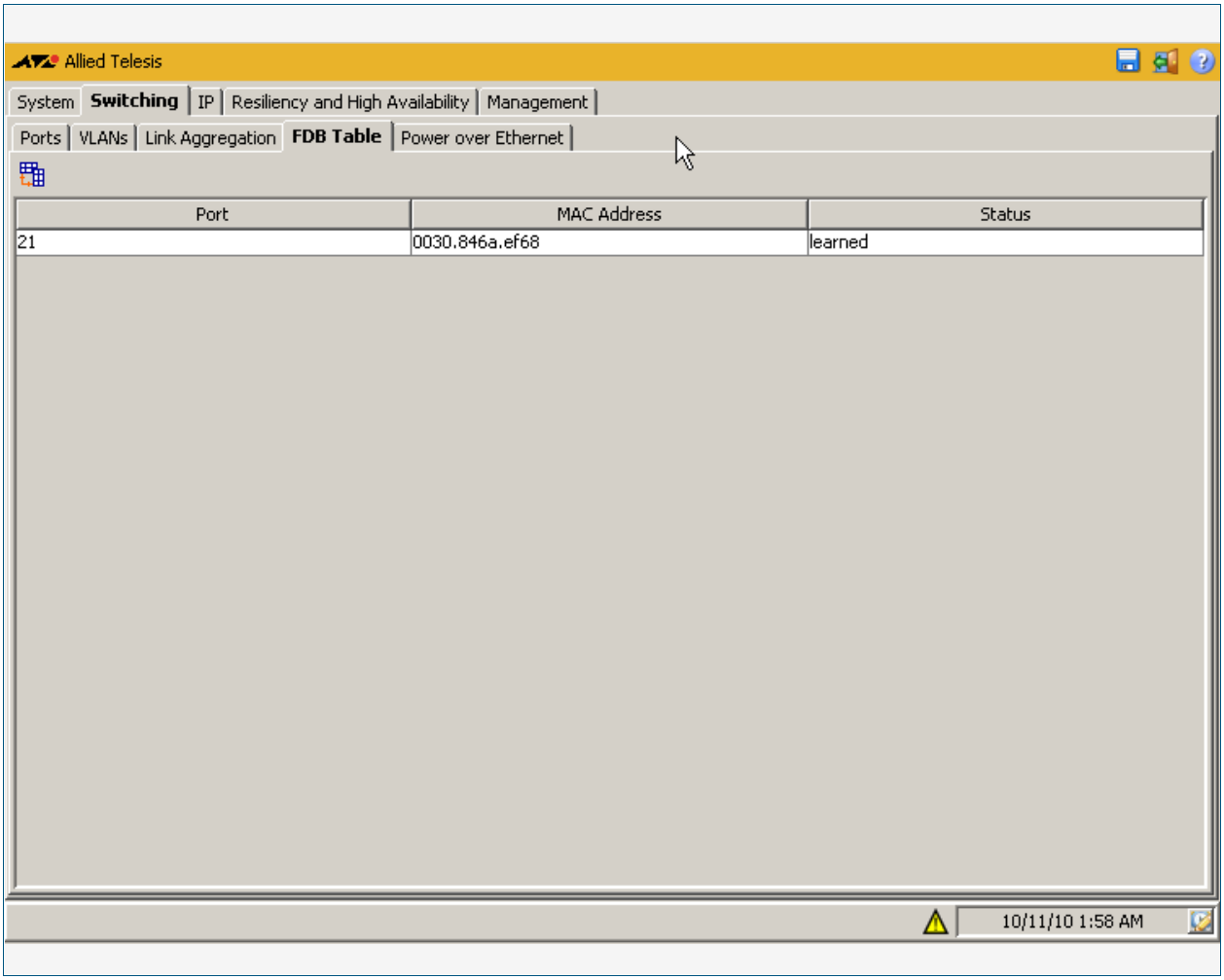
Switching > FDB Table

The **Switching** > **FDB Table** menu tab allows you to view the contents of the Layer 2 Forwarding Database Table.

You can change the FDB Table view to display horizontally or vertically by selecting the table view icon above the FDB Table.

You can also sort or rearrange the display of the FDB Table by Port, MAC Address, or Forwarding Status by selecting the relevant column or by dragging the relevant column respectively.

Menu Tab Example showing the **Switching** > **FDB Table** menu tab:



Description

Label / Field / Button	Description
`FDB Table`	Displays the FDB (Forwarding Database) table for the switch that shows all the available ports, MAC addresses, and port status.

Switching > Power over Ethernet

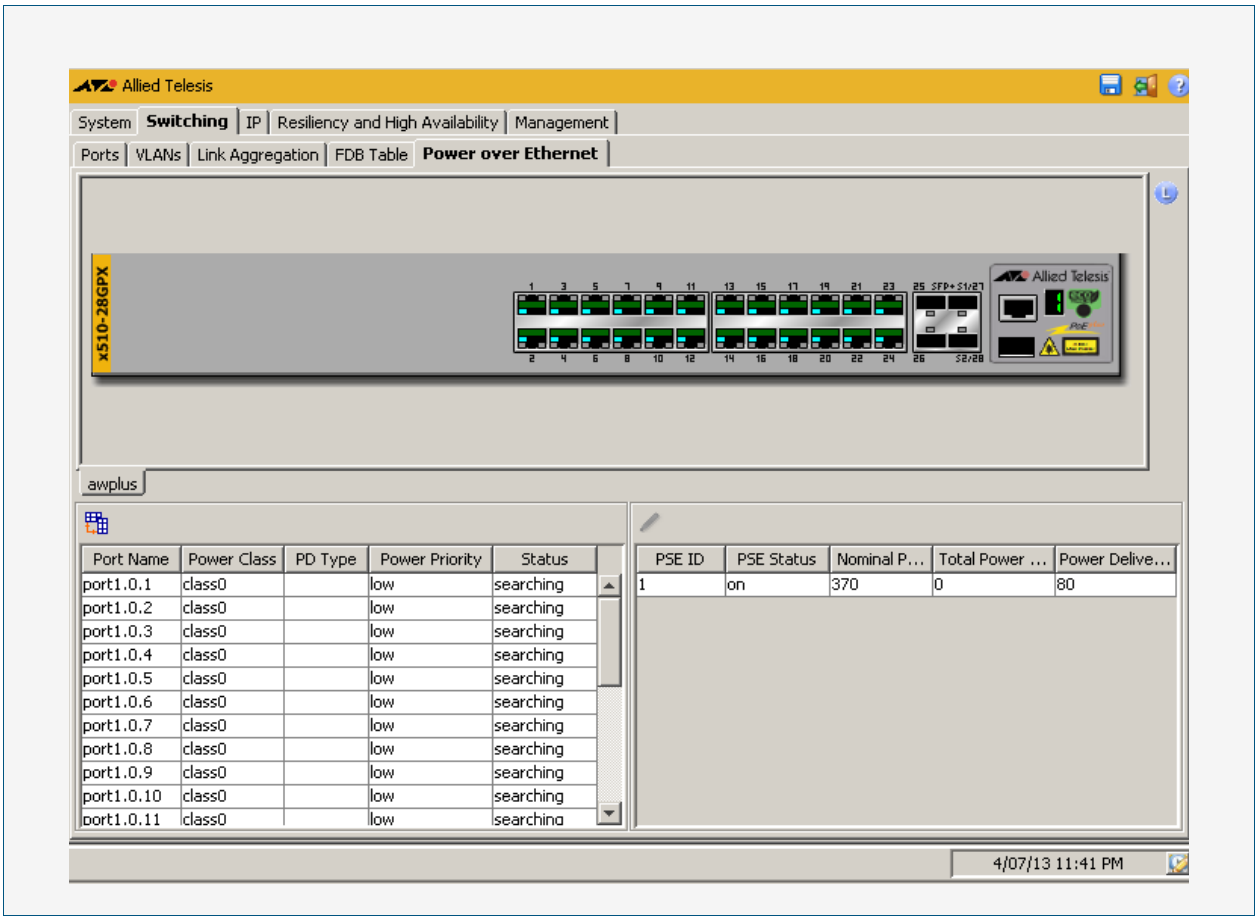
The **Switching** > **Power over Ethernet** menu tab allows you to monitor and configure PoE on your PoE switch. You can monitor PoE status, configure PoE ports and set the PSE power for your switch.

To configure PoE on a port, right-click on the port to view the configuration dialog as shown here **Switching > Power over Ethernet > Configure Port** to enable or disable PoE on a port, set the power priority, and add or edit a PD description.

To configure PoE on the PSE click on the pen icon above the PSE ID label to view the configuration dialog as shown here **Switching > Power over Ethernet > Configure PSE** to set the power delivery threshold for the PSE as a percentage of total nominal power available.

For introductory information about the Power over Ethernet feature on AlliedWare Plus™ see **Chapter 24, Power over Ethernet Introduction**.

Menu Tab Example showing the **Switching** > **Power over Ethernet** menu tab:



Description

Label / Field / Button	Description
`Port Name`	Displays the PoE port(s) in the format `portx.y.z`, where `x` is the device number, `y` is the module number within the device, and `z` is the PoE port number within the module.
`Power Class`	**class0**, **class1**, **class2**, **class3, class4** may be shown according to the class of PD discovered and classified to the IEEE802.3af PD classifications: **class0** devices have 15.4W of power supplied from the PSE. **class1** devices have 4.0W of power supplied from the PSE. **class2** devices have 7.0W of power supplied from the PSE. **class3** devices have 15.4W of power supplied from the PSE. **class4** devices have 30W of power supplied from the PSE (PoE+).
`PD Type`	Adding a PD (Powered Device) description allows the PoE switch to display its function, name, or type of PD connected to the PoE port. Knowing the PD is useful to confirm PD Class power usage.
`Power Priority`	There are three priority levels: **Critical**, **High**, and **Low**. **Critical** is the highest priority level. Ports set to this level are guaranteed power before any ports assigned High or Low priority. **High** is the second highest priority level. Ports set to this level receive power only if all the ports set to Critical are receiving power. **Low** is the lowest priority level. This is the default priority setting. Ports set to this level only receive power if ports set to Critical and ports set to High are receiving power. Note that if there is not enough power for all ports set to the Low priority level then power is provided to the ports based on port number, in ascending order.
`Status`	Displays the current PSE PoE port state, as listed below: **disabled** displays when PoE has been disabled for the PoE port **searching** displays when PoE has been enabled for the PoE port **delivering power** displays when there is a PD connected to a PoE port and power is being supplied from the PSE to the PD via the PoE port **fault** displays when a problem is detected with the PoE device **test** displays when the PoE port is in test mode **other fault** displays when the switch is unable to supply power to the PoE port
`PSE ID`	The ID of the PSE is displayed. The PSE ID is the same number as the stack member number assigned to the PSE for VCStack operation.

Label / Field / Button(cont.)	Description(cont.)
PSE Status	Displays the operational status of the PSU hardware on the PSE (Power Sourcing Equipment): **On** is the PSU as installed in the PSE is switched on. **Off** when the PSU as installed in the PSE is switched off (note that an RPS (Redundant Power Supply) may be connected to the PSE to power PoE instead of the PSE PSU when Off and power is supplied). **Fault** when there is an issue with the PSE PSU hardware.
Nominal Power	Displays the nominal power available from the PSE in watts (W).
Total Power Consumption	Displays the current total power consumption in watts (W) drawn by any connected Powered Devices (PDs) and updates every 5 seconds.
Power Delivery Threshold	This is the level at which the Power Sourcing Equipment (PSE) details that power supplied to all Powered Devices (PDs) has reached a critical level of the nominal power rating for the PSE. The default power delivery threshold is 80% of the nominal PSE power rating.

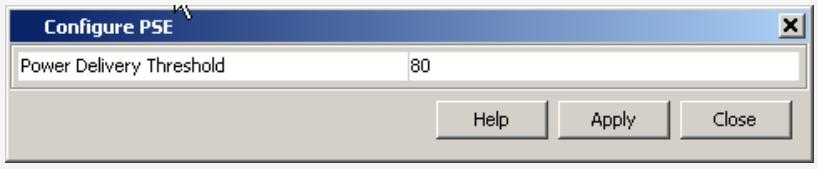
Switching > Power over Ethernet > Configure PSE

The **Switching** > **Power over Ethernet** > **Configure PSE** dialog allows you to configure the power delivery threshold level for the PSE.

For introductory information about the Power over Ethernet feature on AlliedWare Plus™ see **Chapter 24, Power over Ethernet Introduction**.

Configuration Dialog

Example showing the **Switching** > **Power over Ethernet** > **Configure PSE** dialog:



Description

Label / Field / Button	Description
Power Delivery Threshold	This is the level at which the Power Sourcing Equipment (PSE) details that power supplied to all Powered Devices (PDs) has reached a critical level of the nominal power rating for the PSE. The default power delivery threshold is 80% of the nominal PSE power rating.

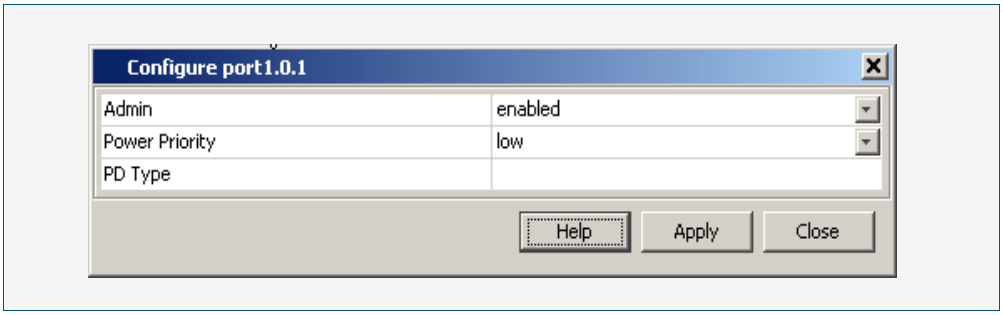
Switching > Power over Ethernet > Configure Port

The **Switching** > **Power over Ethernet** > **Configure Port** dialog allows you to enable or disable PoE on ports, set the power priority for ports, and add or edit PD descriptions.

For introductory information about the Power over Ethernet feature on AlliedWare Plus™ see **Chapter 24, Power over Ethernet Introduction**.

Configuration Dialog

Example showing **Switching** > **Power over Ethernet** > **Configure Port** dialog:

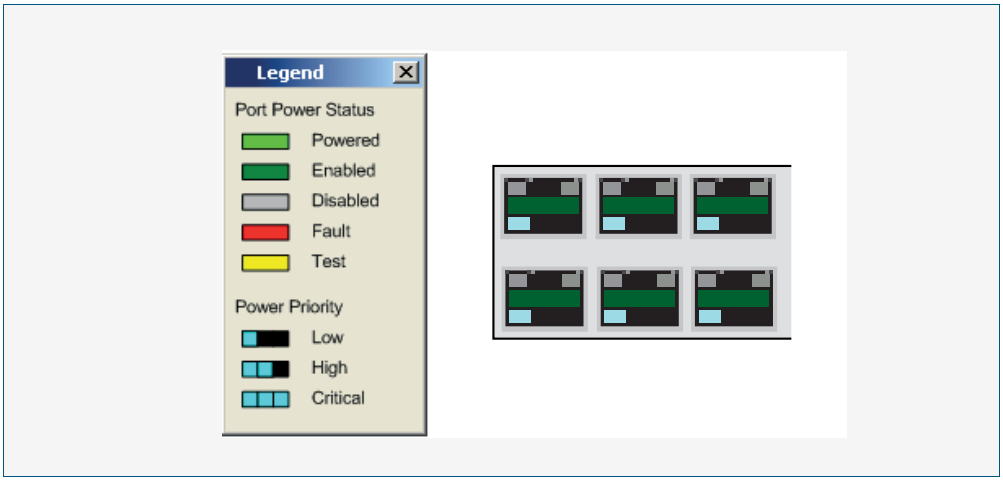


To display the Configure port sub screen, select a port from the front panel image, then right-click your mouse on this port.

Description

Label / Field / Button	Description
`Admin`	Displays the administrative state of PoE on a PoE port, either **Enabled** or **Disabled**.
`Power Priority`	There are three priority levels: **Critical**, **High**, and **Low**. **Critical** is the highest priority level. Ports set to this level are guaranteed power before any ports assigned High or Low priority. **High** is the second highest priority level. Ports set to this level receive power only if all the ports set to Critical are receiving power. **Low** is the lowest priority level. This is the default priority setting. Ports set to this level only receive power if ports set to Critical and ports set to High are receiving power. Note that if there is not enough power for all ports set to the Low priority level then power is provided to the ports based on port number, in ascending order.
`PD Type`	Adding a PD description allows the PoE switch to display its function, name, or type of PD connected to the PoE port. Knowing the PD is useful to confirm PD Class power usage.

Legend Selected by **Switching > Power over Ethernet > Legend** the legend shows the PoE status for each port. The following example shows the **Port Power Status and Power Priority for ports on the switch front panel LEDs**:



To display the Legend, select - from the top right hand side of the front panel display - the small icon that has a white L within a blue background.

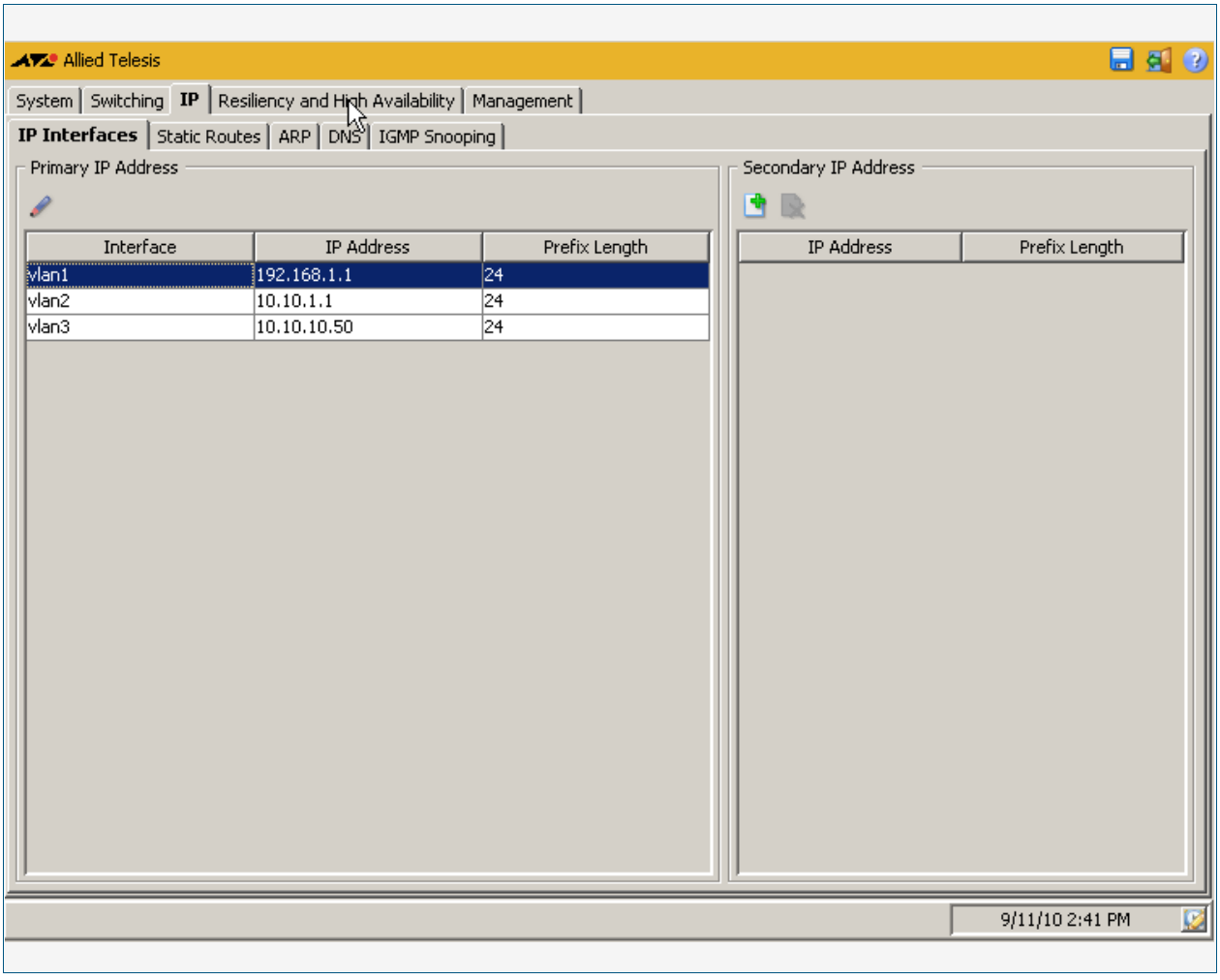
IP > IP Interfaces

The **IP > IP Interfaces** menu tab allows you to view and specify the Primary and Secondary IP Addresses for VLAN and management port interfaces.

Note You may only define a Secondary IP Address for an interface after first defining its Primary IP Address.

- Select an interface then use the pen shaped icon under the Primary IP Address label to configure an IP address. You can delete an assigned Primary IP Address in the Configure Primary IP Address dialog as displayed after selecting the icon under Primary IP Address.
- Select an interface with a Primary IP Address already defined to configure a Secondary IP Address, using the **+** icon under the Secondary IP Address label.
- Remove a selected Secondary IP Address using the **x** icon under the Secondary IP Address label.

Menu Tab Example showing the **IP > IP Interfaces** menu tab:



Description

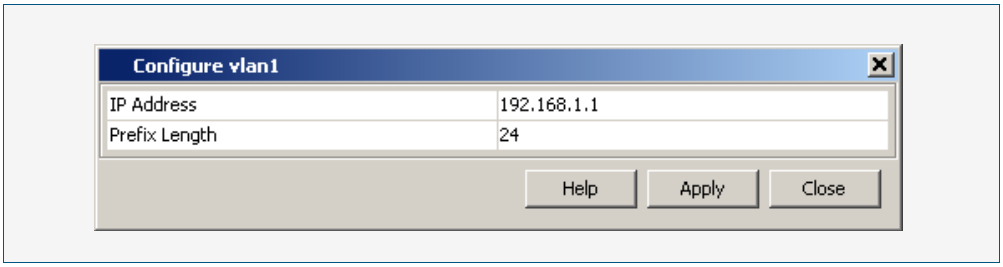
Label / Field / Button	Description
`Primary IP Address`	Displays and configures primary IP addressing for VLANs and management port interfaces that are defined on the switch and assigned to ports.
`Secondary IP Address`	Displays and configures secondary IP addressing for VLANs and management port interfaces that are defined on the switch and assigned to ports.

IP > IP Interfaces > Configure Primary IP Address

The **IP > IP Interfaces > Configure Primary IP Address** dialog allows you to configure a primary address with an IP address and a prefix length for the selected interface.

Configuration Dialog

Example showing the **IP > IP Interfaces > Configure Primary IP Address** dialog:



Description

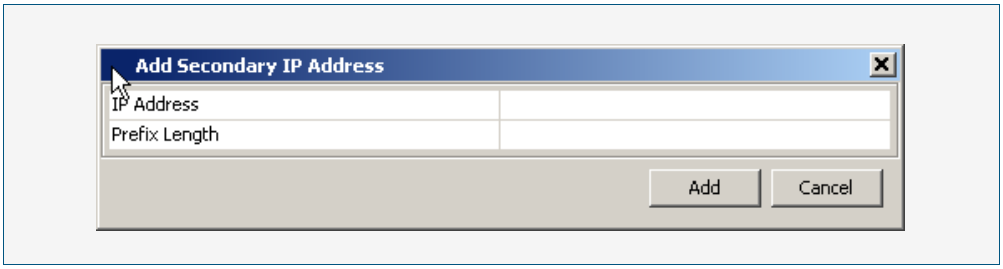
Label / Field / Button	Description
`IP Address`	Enter or remove an IPv4 Address in dotted decimal notation (i.e. A.B.C.D)
`Prefix Length`	Enter the Prefix for the IPv4 Address with the integer as used for slash notation (i.e. 24 instead of /24), not dotted decimal notation (i.e. 255.255.255.0)

IP > IP Interfaces > Add Secondary IP Address

The **IP** > **IP Interfaces** > **Add Secondary IP Address** dialog allows you to add a secondary address with an IP address and a prefix length for the selected interface (only if the selected interface already has a primary address configured).

Configuration Dialog

Example showing the **IP > IP Interfaces > Add Secondary IP Address** dialog:



Description

Label / Field / Button	Description
`IP Address`	Enter an IPv4 Address in dotted decimal notation (i.e. `A.B.C.D`).
`Prefix Length`	Enter the Prefix for the IPv4 Address with the integer as used for slash notation (i.e. `24` instead of `/24`), not dotted decimal notation (i.e. `255.255.255.0`).

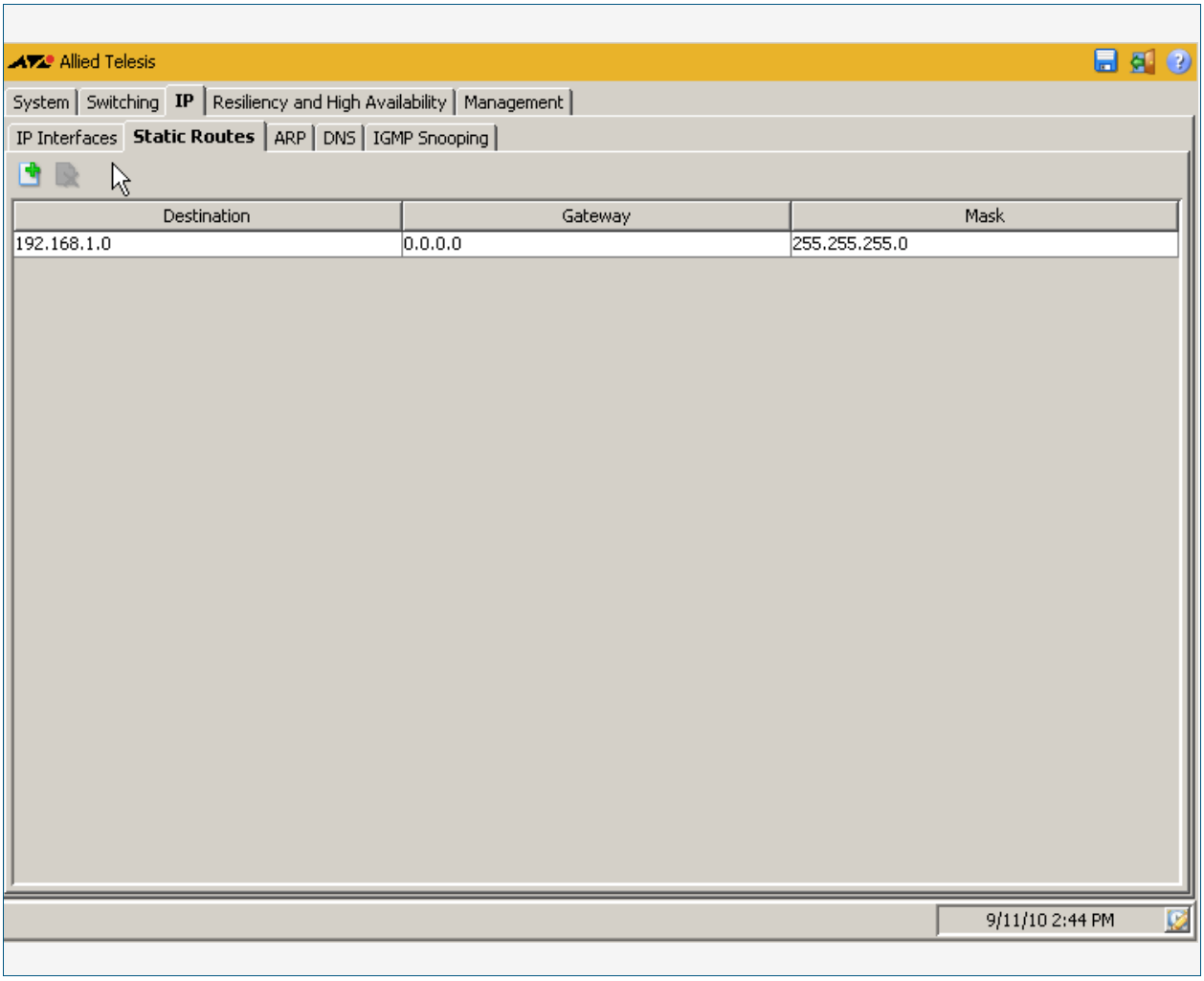
IP > Static Routes

The **IP** > **Static Routes** menu tab allows you to view, add, and delete static IP routes configured on the switch. Static routes are specified using destination IP addresses, masks, and gateways.

You can also sort or rearrange the display of the Static Routes by Destination, Gateway, or Mask by selecting the relevant column or by dragging the relevant column respectively.

- Selecting the **+** icon allows you to define a Static Route specifying destination and gateway IPv4 addresses with a dotted decimal format subnet mask.
- Selecting the **x** icon allows you to delete a defined Static Route.

Menu Tab Example showing the **IP** > **Static Routes** menu tab:



Description

Label / Field / Button	Description
`Static Routes / Destination`	The IPv4 address of the destination subnet address.

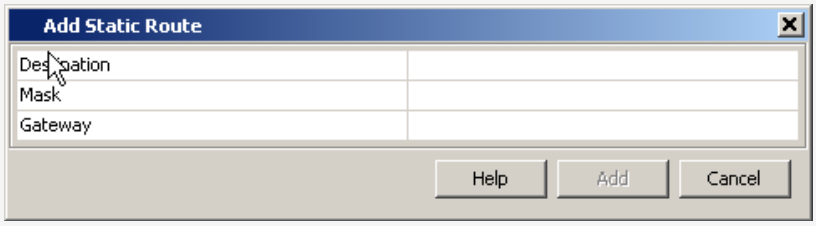
Label / Field / Button	Description(cont.)
`Static Routes / Gateway`	The IPv4 address of the gateway device.
`Static Routes / Mask`	The subnet mask in dotted decimal notation (e.g. 255.255.255.0 instead of the slash notation /24).

IP > Static Routes > Add Static Route

The **IP > Static Routes > Add Static Route** dialog allows you to add a static IP routes on the switch. Static routes are specified using destination IP addresses, masks, and gateways.

Configuration Dialog

Example showing the **IP > Static Routes > Add Static Route** dialog:



Description

Label / Field / Button	Description
`Destination`	Enter the IPv4 address of the destination subnet address.
`Mask`	Enter the subnet mask in dotted decimal notation (e.g. 255.255.255.0 instead of the slash notation /24).
`Gateway`	Enter the IPv4 address of the gateway device.

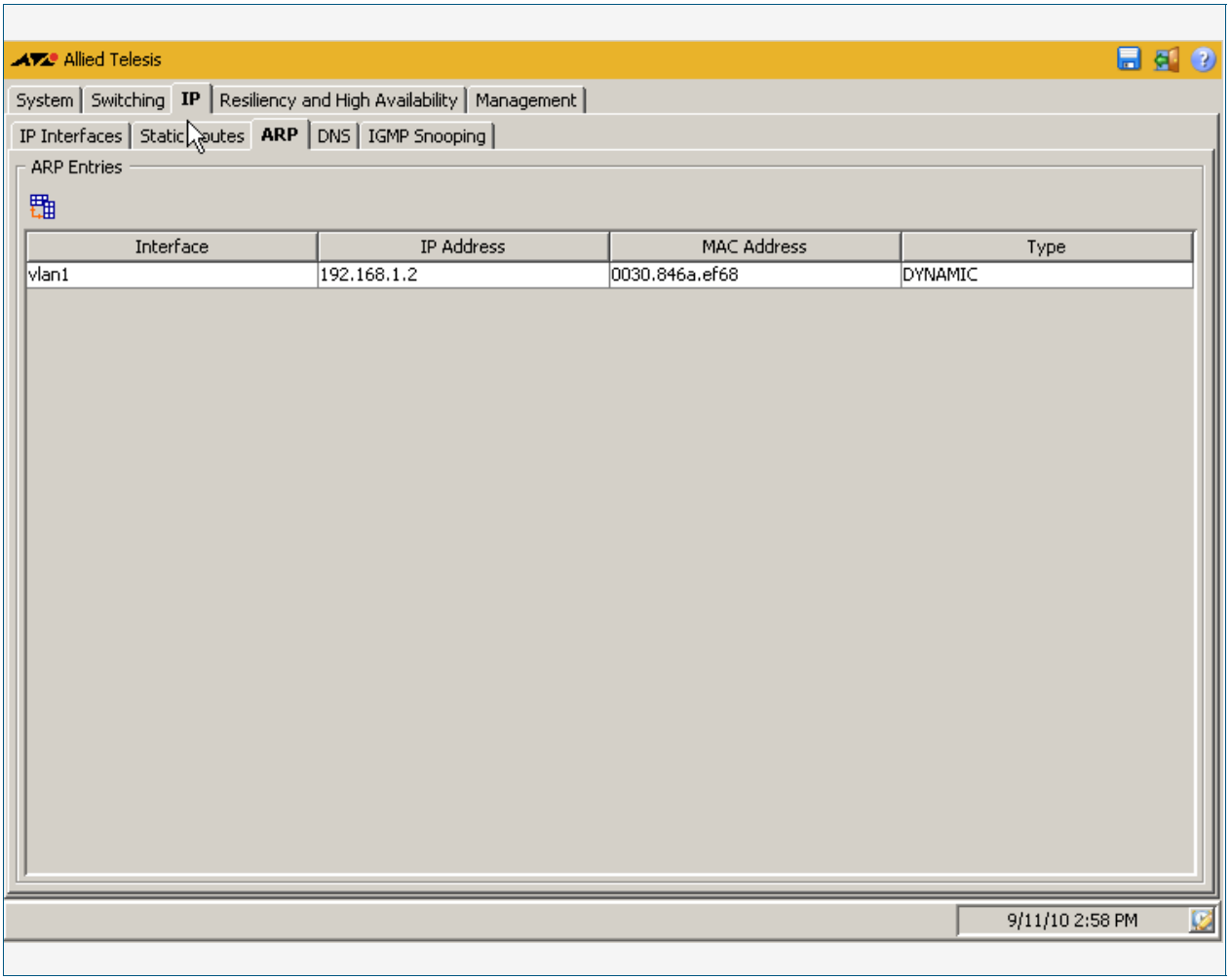
IP > ARP

The **IP** > **ARP** menu tab allows you to view the contents of the ARP (Address Resolution Protocol) Table.

You can change the ARP Entries view to display horizontally or vertically by selecting the table view icon above the ARP Entries.

You can also sort or rearrange the display of the ARP Entries by Interface, IP Address, MAC Address, or Type by selecting the relevant column or by dragging the relevant column respectively.

Menu Tab Example showing the **IP** > **ARP** menu tab:



Description

Label / Field / Button	Description
`ARP Entries / Interface`	Interface over which the switch is accessed, usually a VLAN.
`ARP Entries / IP Address`	IP address of the network device this ARP entry maps to.

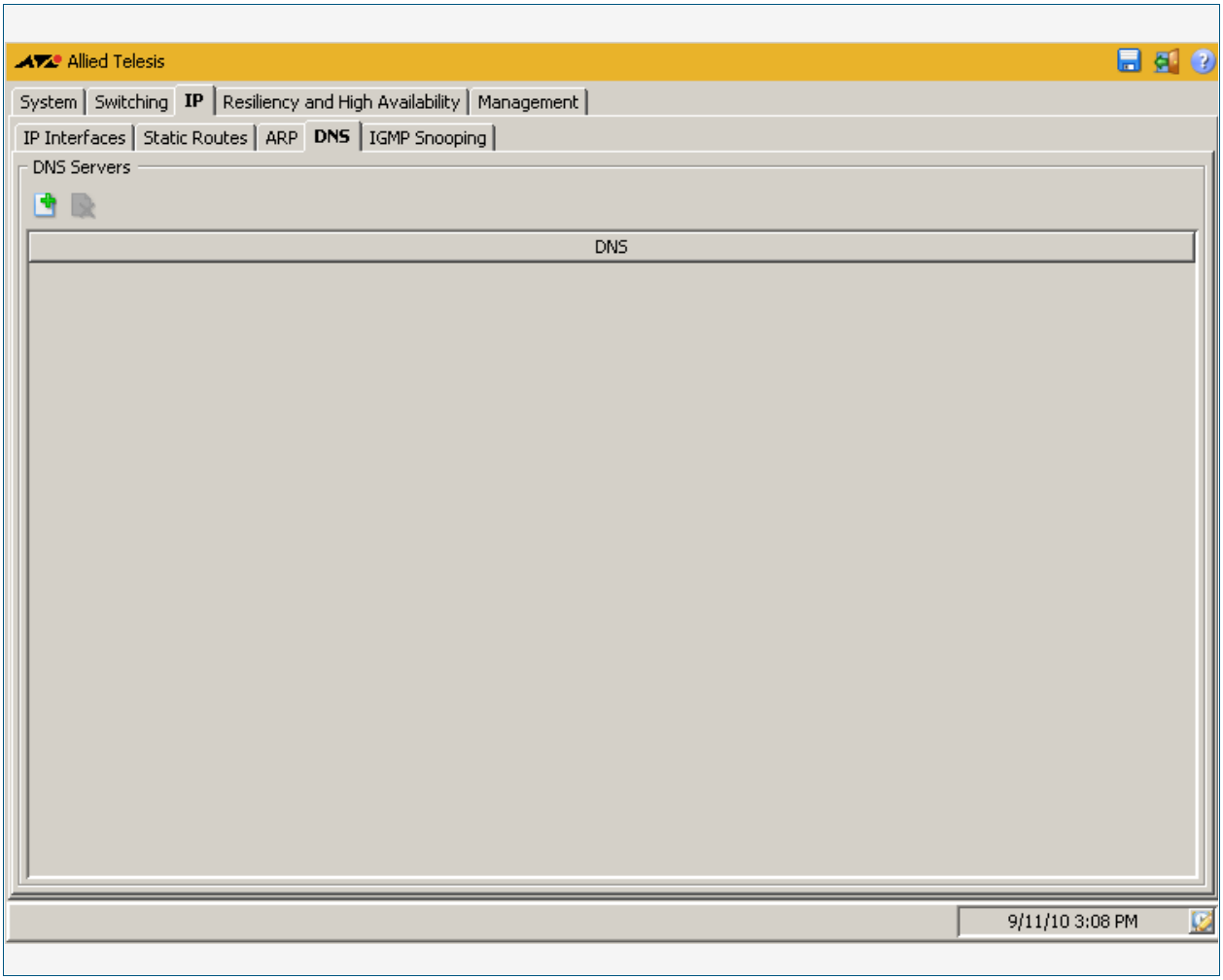
Label / Field / Button	Description(cont.)
ARP Entries / MAC Address	Hardware address of the switch in hexadecimal format HHHH.HHHH.HHHH.
ARP Entries / Type	Indicates whether the ARP entry is a Static or Dynamic ARP entry. Static ARP entries are added and dynamic ARP entries are learned.

IP > DNS

The **IP** > **DNS** menu tab allows you to display and configure DNS (Domain Name System) server entries for the switch.

- Selecting the **+** icon allows you to define a DNS Server specifying the IPv4 address.
- Selecting the **x** icon allows you to delete a defined DNS Server.

Menu Tab Example showing the **IP** > **DNS** menu tab:



Description

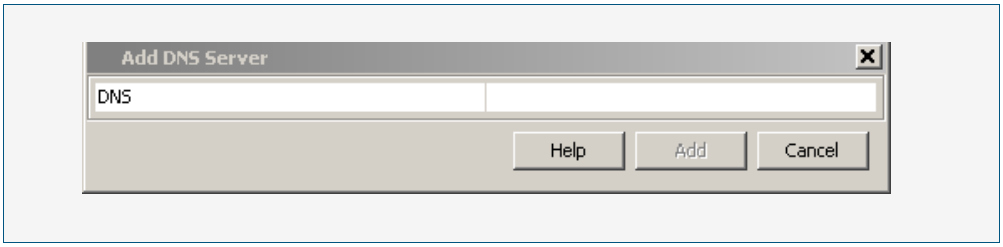
Label / Field / Button	Description
DNS	DNS Server IPv4 address.

IP > DNS > Add DNS Server

The **IP** > **DNS** > **Add DNS Server** dialog allows you to add DNS (Domain Name System) server entries for the switch.

Configuration Dialog

Example showing the **IP** > **DNS** > **Add DNS Server** dialog:



Description

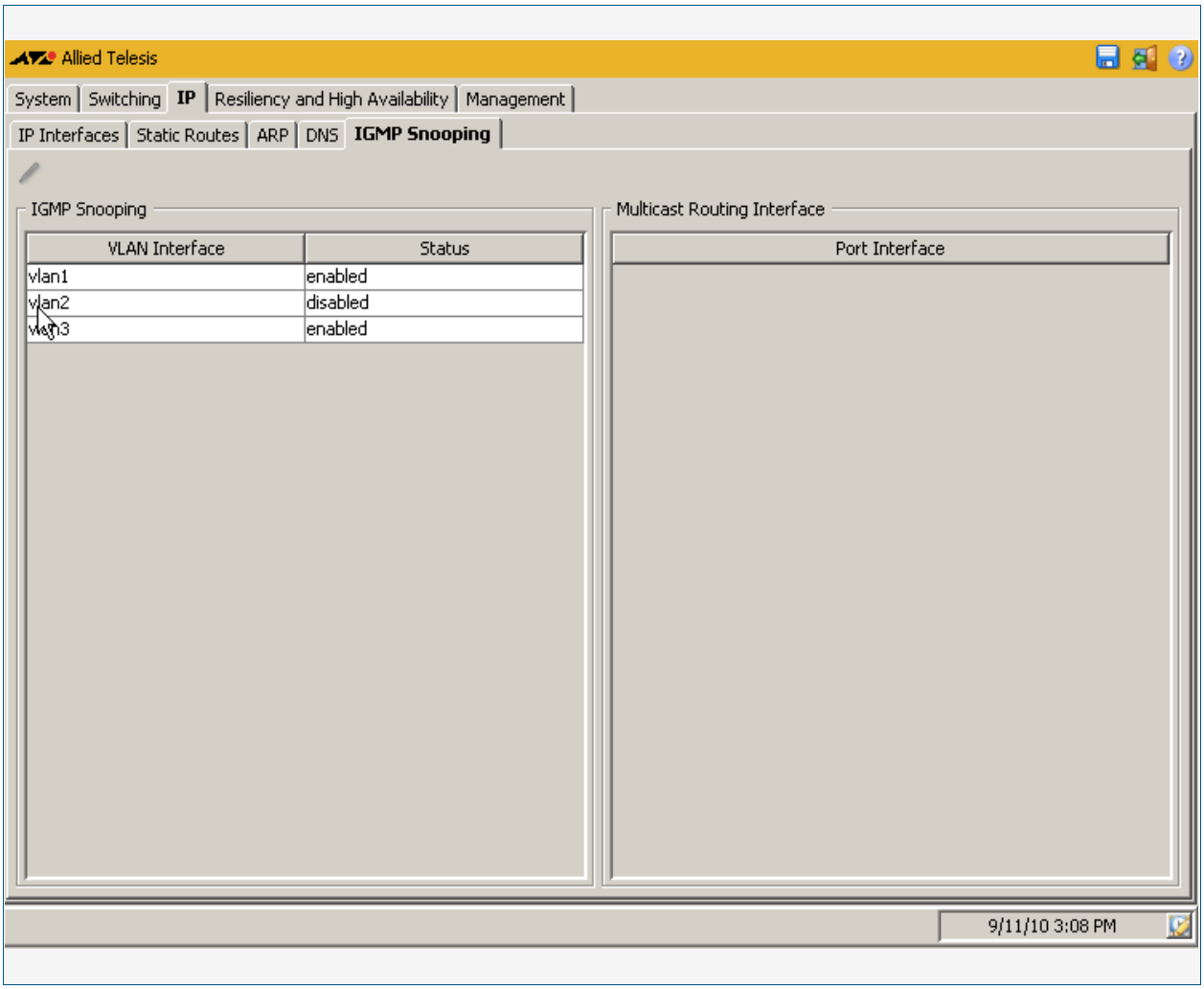
Label / Field / Button	Description
DNS	Enter an IPv4 address in dotted decimal notation (i.e. A.B.C.D) for the DNS (Domain Name System) Server you want to specify.

IP > IGMP Snooping

The **IP > IGMP Snooping** menu tab displays basic IGMP Snooping and Multicast Routing Interface information.

You can also configure IGMP Snooping on individual ports by selecting the VLAN interface that the port is a member of then clicking on the pen shaped icon to display the Configure IGMP Snooping dialog, where you can enable or disable IGMP snooping on desired ports.

Menu Tab Example showing the **IP > IGMP Snooping** menu tab:



Description

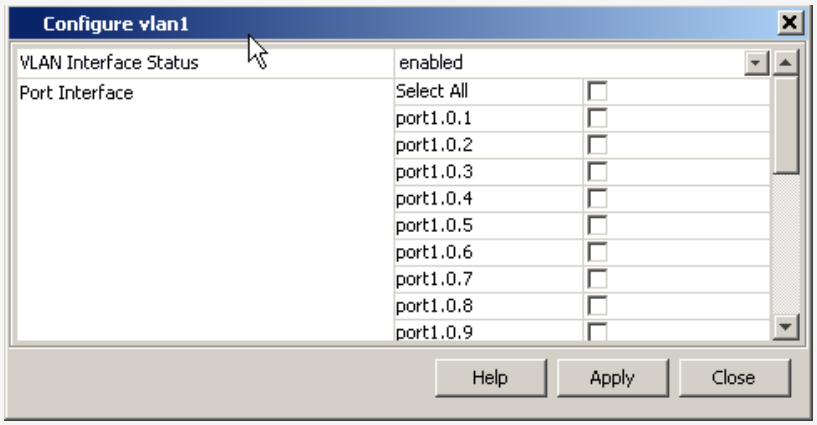
Label / Field / Button	Description
`IGMP Snooping / IGMP Snooping`	Displays and configures IGMP Snooping for a specified VLAN interface.
`IGMP Snooping / Multicast Routing Interface`	Displays and configures the specified port in the VLAN as a multicast router for IGMP Snooping.

IP > IGMP Snooping > Configure Interface

The **IP > IGMP Snooping > Configure Interface** dialog allows you to configure IGMP Snooping on individual ports. First select the VLAN interface that the port is a member of then enable or disable IGMP snooping on desired ports from this dialog.

Configuration Dialog

Example showing the **IP > IGMP Snooping > Configure Interface** dialog:



Description

Label / Field / Button	Description
`VLAN Interface Status`	Displays whether the selected VLAN is enabled or disabled.
`Port Interface`	Select the 'check box' for a given port to allocate it to the VLAN.

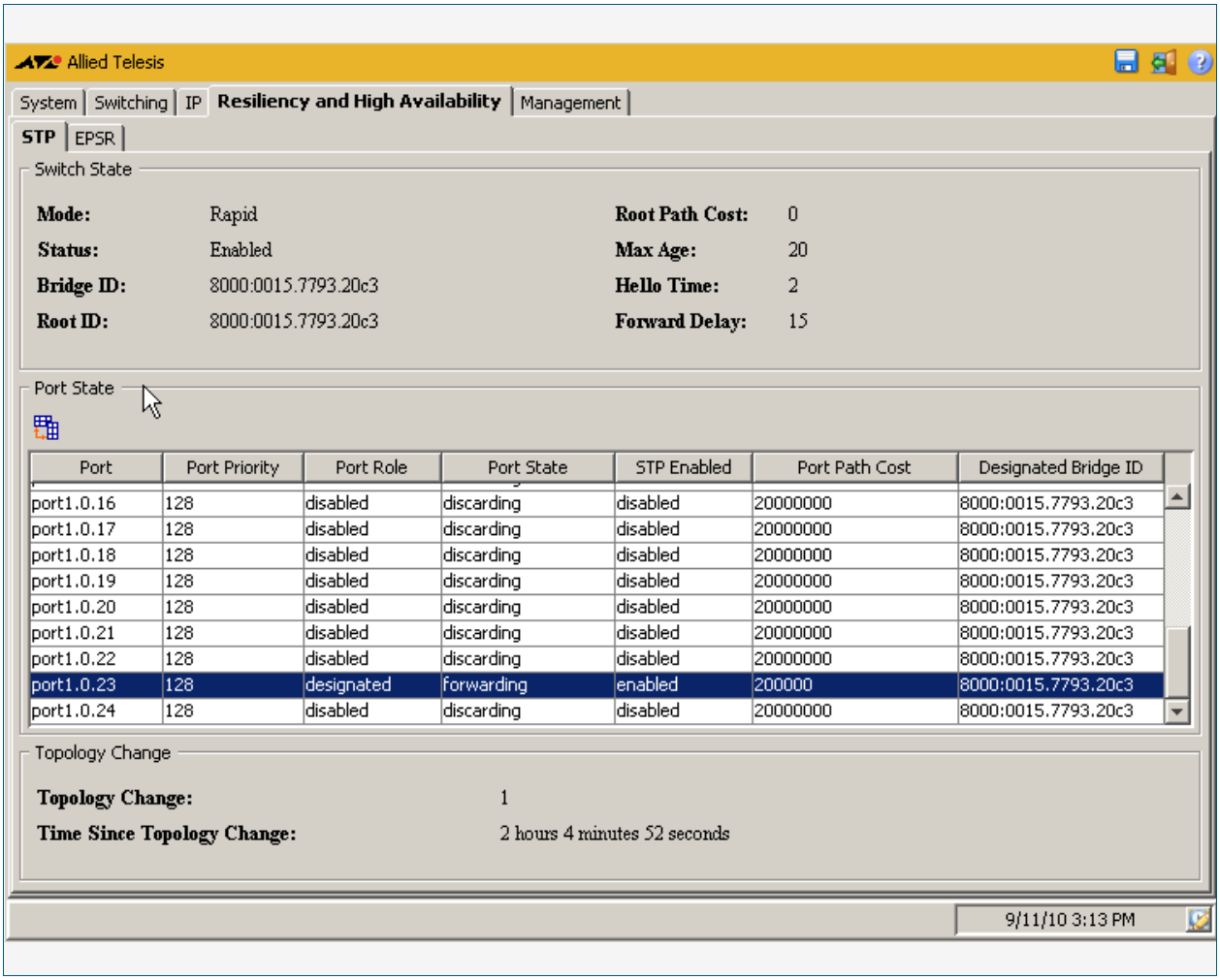
Resiliency and High Availability > STP

The **Resiliency and High Availability > STP** menu tab allows you to view the configuration and status of spanning tree data: for the switch as a whole and for each port.

You can also sort or rearrange the display of the Port State table by Port, Port Priority, Port State, Port Role, STP Enabled, Port Path Cost, or Designated Bridge ID by selecting the relevant column or by dragging the relevant column respectively.

Note STP is not configurable through the GUI. Refer to the relevant STP chapters in the AlliedWare Plus™ Software Reference to configure STP using the CLI instead.

Menu Tab Example showing the **Resiliency and High Availability > STP** menu tab:



Description: Switch State

Label / Field / Button	Description
`Switch State / Mode`	Spanning Tree Mode displayed: STP (Spanning Tree Protocol) or Rapid (Rapid Spanning Tree Protocol - RSTP) or Multiple (Multiple Spanning Tree Protocol - MSTP)
`Switch State / Status`	Status of the Spanning Tree Mode: enabled or disabled.
`Switch State / Bridge ID`	Bridge ID, comprising the port priority followed by its MAC address.
`Switch State / Root ID`	Root Bridge ID, comprising the root priority followed by its MAC address.
`Switch State / Root Path Cost`	Sum of the costs for each path between the bridge port and the root bridge.
`Switch State / Max Age`	Time in seconds that the dynamic spanning tree configuration information is stored in the switch before it is discarded.
`Switch State / Hello Time`	Time in seconds between the transmission of switch spanning tree configuration information, when the switch is the Root Bridge of the spanning tree or is trying to become the Root Bridge.
`Switch State / Forward Delay`	Time in seconds to control how fast a port changes its spanning tree state when moving towards the forwarding state. This value is used only when the switch is acting as the root bridge. Note that Forward Delay, Max Age, and Hello Time are interrelated.

Description: Port State

Label / Field / Button	Description
`Port State / Port`	Switch port number in the format 'portX.Y.Z' where X is the switch, Y is the XEM, and Z is the individual switch port number.
`Port State / Port Priority`	The lower the port priority, the higher the likelihood of the port becoming part of the active network topology.
`Port State / Port Role`	Displays the port role as configured in the CLI with ‘spanning-tree’ commands, and shows either ‘rootport’, ‘backup’, ‘disabled’ or ‘designated’ port roles.
`Port State / Port State`	Displays the spanning tree state for the port as configured in the CLI with 'spanning-tree' commands. Indicates spanning tree states of: disabled, blocking, listening, learning, and forwarding.
`Port State / STP Enabled`	Displays whether spanning-tree is enabled or disabled. Spanning tree is enabled by default.
`Port State / Port Path Cost`	The cost of a path for the port that determines the total cost path. The lower the total cost, the higher the priority of the path.
`Port State / Designated Bridge ID`	The unique parent for each bridge that connects it to the next LAN on the path towards the root bridge.

Description: Topology Change

Label / Field / Button	Description
`Topology Change / Topology Change`	The number of STP Topology Changes that have occurred since the switch was rebooted.
`Topology Change / Time Since Topology Change`	The time in hours and seconds since the previous STP Topology Change occurred.

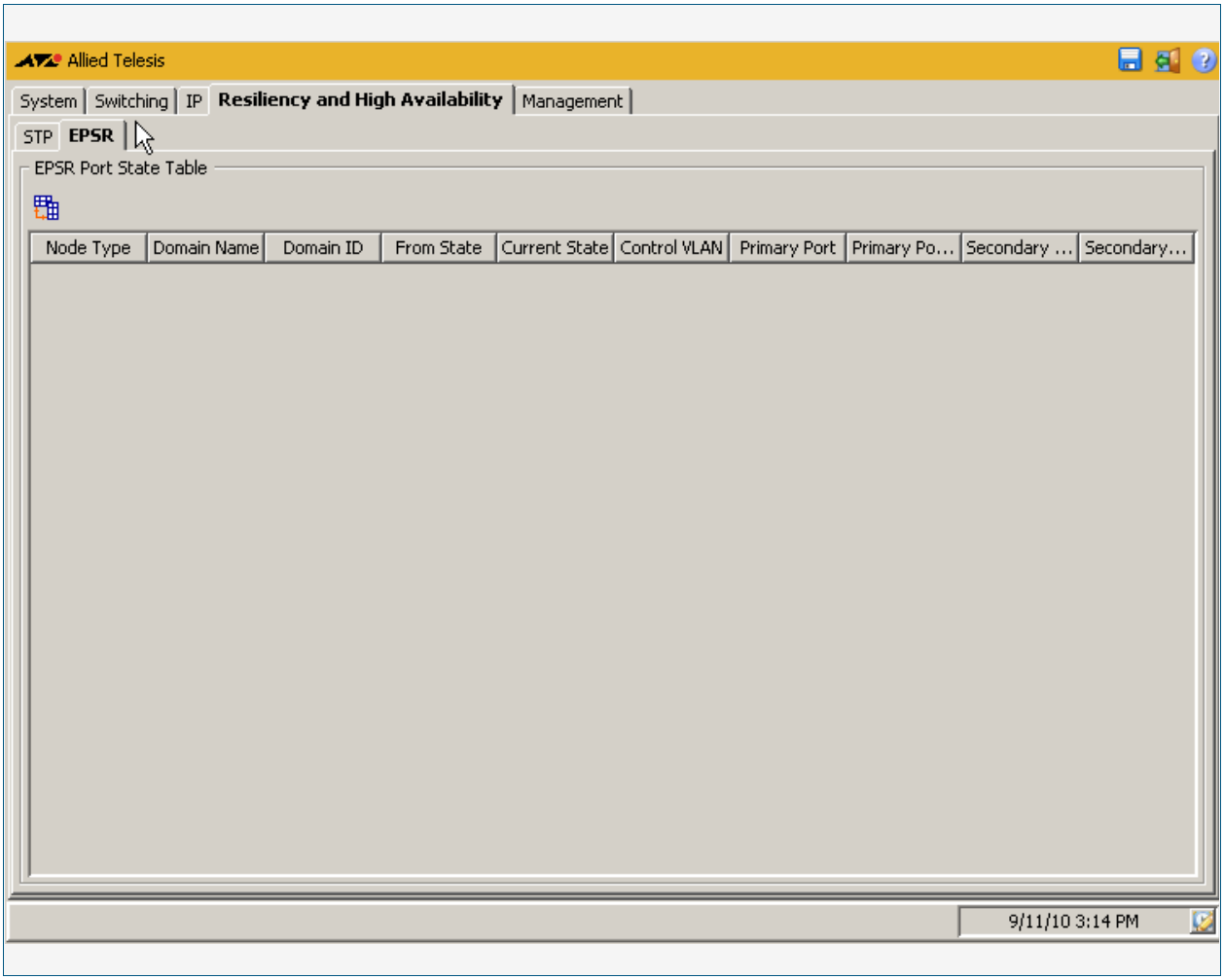
Resiliency and High Availability > EPSR

The **Resiliency and High Availability** > **EPSR** menu tab allows you to display the properties and status of any EPSR domains configured on the switch.

You can also sort or rearrange the display of the EPSR Port State table by Node Type, Domain Name, Domain ID, From State, Current State, Control VLAN, Primary Port, Primary Port Status, Secondary Port, or Secondary Port Status by selecting the relevant column or by dragging the relevant column respectively.

Note EPSR is not configurable through the GUI. Refer to the relevant EPSR chapters in the AlliedWare Plus™ Software Reference to configure EPSR using the CLI instead.

Menu Tab Example showing the **Resiliency and High Availability** > **EPSR** menu tab:



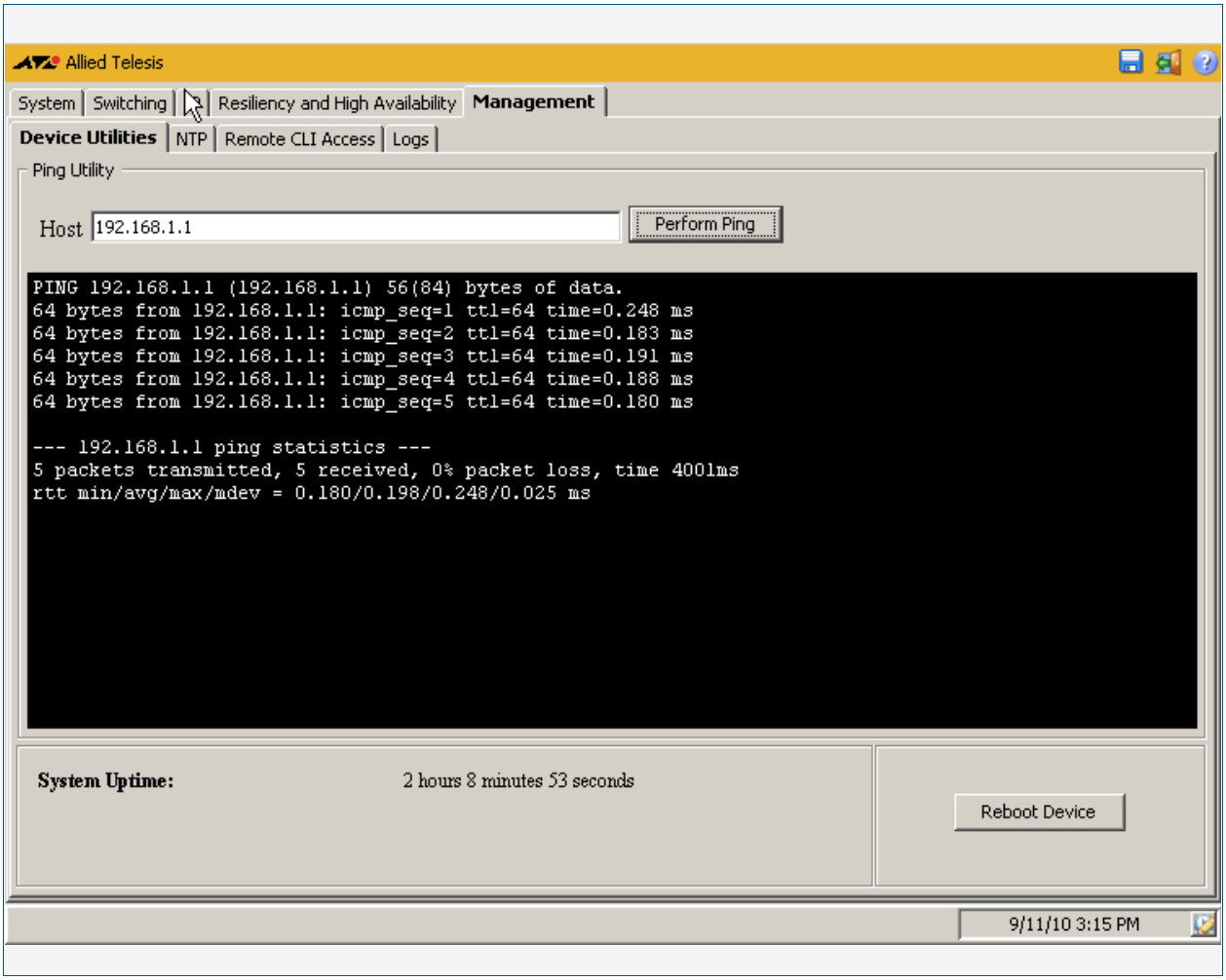
Description

Label / Field / Button	Description
EPSR Port State Table / Node Type	Displays master or transit node as configured in the CLI with the '**epsr mode**' command.
EPSR Port State Table / Domain Name	Displays the domain name. A set of instances across a ring is called a domain.
EPSR Port State Table / Domain ID	Displays the assigned domain number for the domain name.
EPSR Port State Table / From State	Displays the From EPSR state as configured in the CLI with the '**epsr state**' commands.
EPSR Port State Table / Current State	Displays the Current EPSR state as configured in the CLI with the '**epsr state**' commands.
EPSR Port State Table / Control VLAN	Displays the control VLAN as configured in the CLI with the '**epsr mode controlvlan**' command.
EPSR Port State Table / Primary Port	Displays the master node primary port interface name as configured in the CLI with the '**epsr mode primaryport**' command.
EPSR Port State Table / Primary Port Status	Displays the master node primary port interface status: up or down.
EPSR Port State Table / Secondary Port	Displays the assigned secondary port interface name.
EPSR Port State Table / Secondary Port Status	Displays the assigned secondary port interface status: up or down.

Management > Device Utilities

The **Management** > **Device Utilities** menu tab allows you to perform pings and reboot the switch from the GUI.

Menu Tab Example showing the **Management** > **Device Utilities** menu tab:



Description

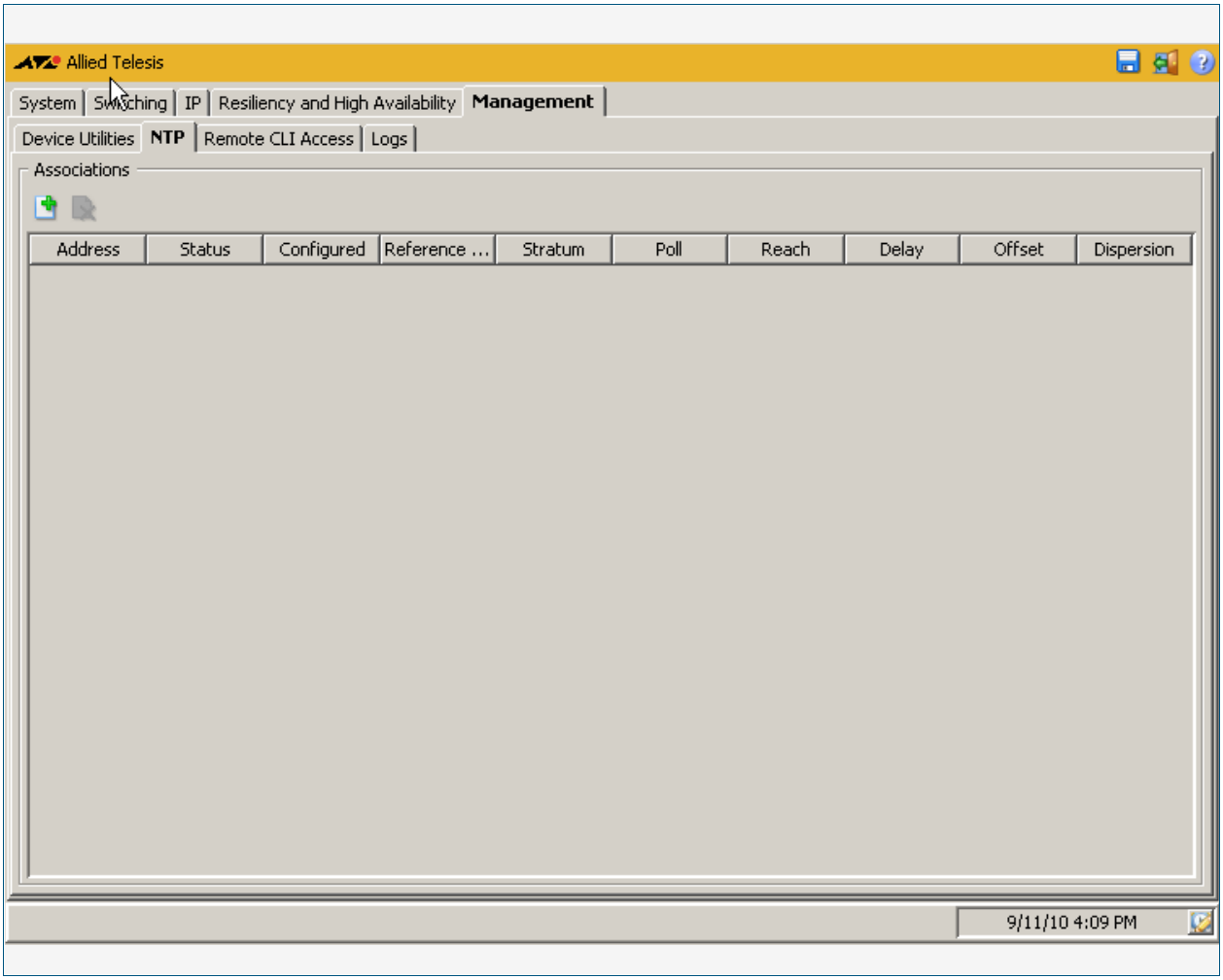
Label / Field / Button	Description
`Ping Utility / Host`	Enter the IPv4 address or the URL that you want to ping in this field.
`Ping Utility / Perform Ping`	Select this button to ping the IPv4 address or URL that you entered in the Host field.
`System Uptime`	Displays the elapsed time since the last reboot in hours, minutes, and seconds.
`Reboot Device`	Select this button to reboot your switch. You will need to login to the GUI again after you reboot your switch. Rebooting closes all Telnet / SSH / SNMP sessions on your switch.

Management > NTP

The **Management** > **NTP** menu tab allows you to display and configure Network Time Protocol (NTP) peer configurations on the switch.

- Selecting the **+** icon allows you to add an NTP association.
- Selecting the **x** icon allows you to delete an NTP association.

Menu Tab Example showing the **Management** > **NTP** menu tab:



Description

Label / Field / Button	Description
`Associations / Address`	The NTP peer or NTP server IPv4 address.
`Associations / Status`	Indicates association status, and displays '**master(synced)**'. '**master(unsynced)**', '**selected**', '**candidate**', '**configured**', or '**unknown**'.
`Associations / Configured`	Indicates if the association is configured or not, and displays '**configured**' or '**dynamic**'.

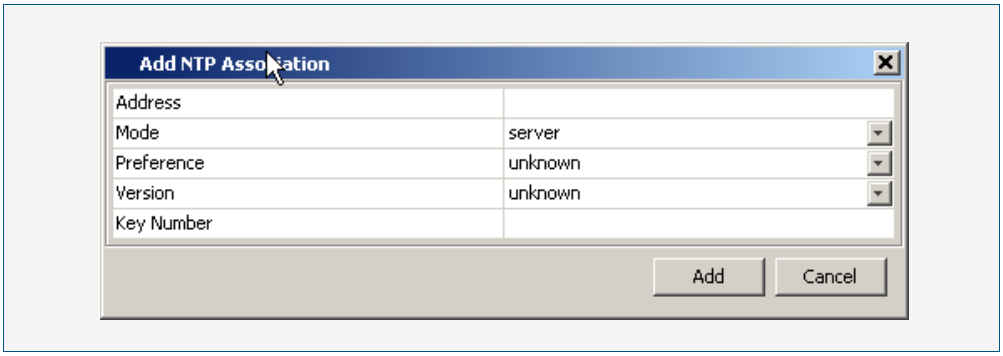
Label / Field / Button(cont.)	Description(cont.)
`Associations / Reference Clock`	The IPv4 address for the reference clock.
`Associations / Stratum`	The number of hops between the server and the accurate time source.
`Associations / Poll`	The time between NTP requests from the device to the server.
`Associations / Reach`	Shows whether or not the NTP server responded to the last request, which indicates the reachability of the NTP peer.
`Associations / Delay`	The round trip delay between the device and the server.
`Associations / Offset`	The difference between the device clock and the server clock, relative to the server clock, in milliseconds.
`Associations / Dispersion`	The lowest measure of error associated with peer offset based on delay.

Management > NTP > Add NTP Association

The **Management > NTP > Add NTP Association** dialog allows you configure Network Time Protocol (NTP) peer configurations on the switch.

Configuration Dialog

Example showing the **Management > NTP > Add NTP Association** dialog:



Description

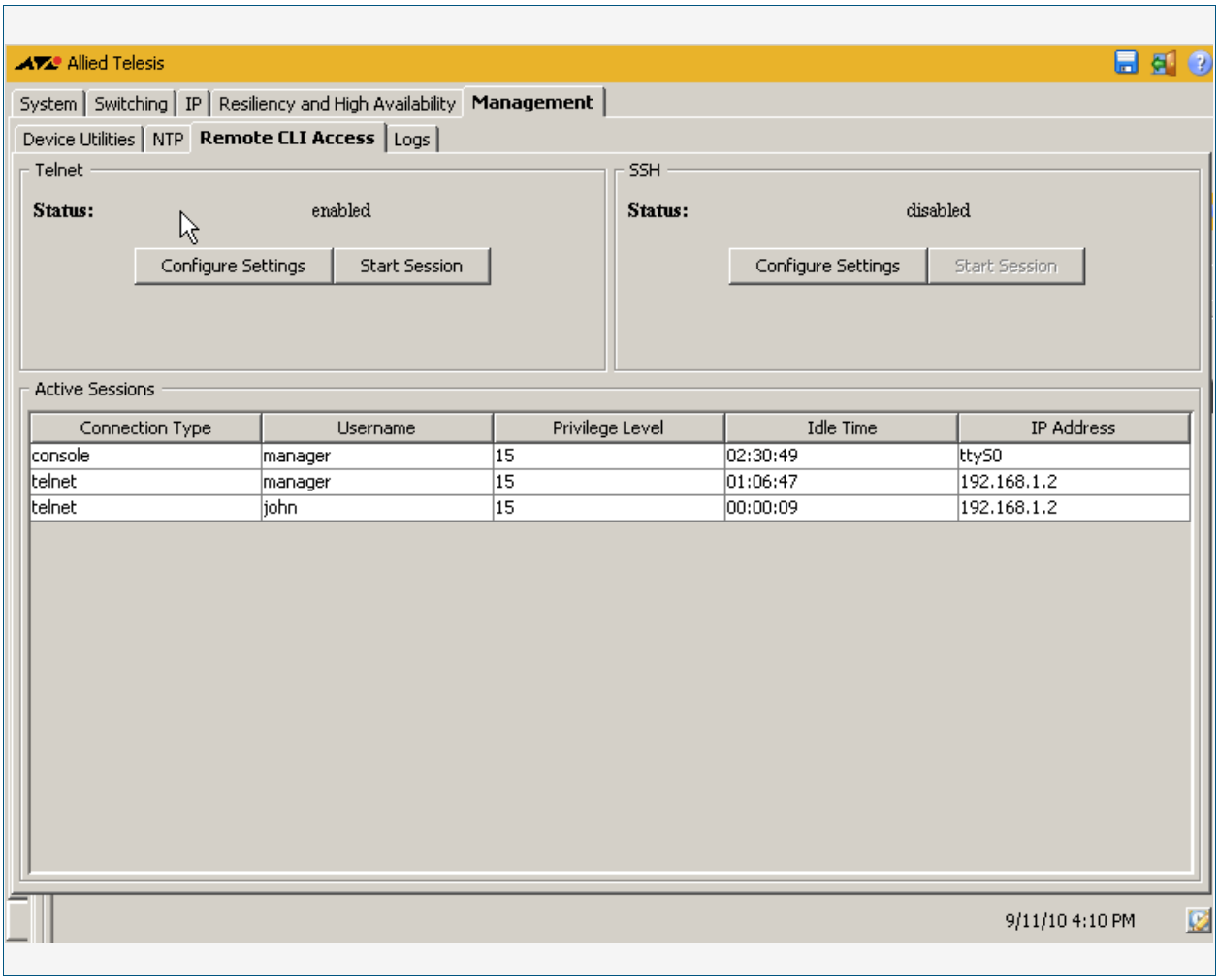
Label / Field / Button	Description
`Address`	Enter the NTP IPv4 address for the NTP peer or NTP server used.
`Mode`	Select one of the `server` or `peer` options from the drop down list to specify the NTP Mode used. When using NTP server mode, the NTP server will not accept updates from clients for updates to the server's time settings. The NTP server is configured to synchronize the NTP clients. When using NTP peer mode, each device shares its time information with the other, and each device can also provide time synchronization to the other.
`Preference`	Select one of the `unknown`, `not preferred`, or `preferred` options from the drop down list to specify the NTP Preference used. NTP Preference is used to configure an NTP server, so the NTP server is given preference to synchronize the NTP clients.
`Version`	Select one of the `unknown`, `version 1`, `version 2`, `version 3`, or `version 3, version 4` options from the drop down list to specify the NTP Version used.
`Key Number`	Enter the NTP Key Number for NTP authentication, which allows NTP to authenticate the associations with other systems for security purposes. The NTP Key Number is an integer in the range <1-4294967295>. The MD5 (Message-Digest algorithm 5) key type is supported to encrypt the NTP Key Number used for authentication.

Management > Remote CLI Access

The **Management** > **Remote CLI Access** menu tab allows you to enable, disable and configure Telnet and SSH.

You can create Telnet or SSH connections to the switch, and you can view a list of all current active CLI sessions on the switch from this tab.

Menu Tab Example showing the **Management** > **Remote CLI Access** menu tab:



Description Telnet

Label / Field / Button	Description
`Telnet / Status`	Displays the current Telnet status, either '`enabled`' or '`disabled`'.
`Telnet / Configure Settings`	Configures the Telnet Status. Select '`enabled`' or '`disabled`' to configure the status of the Telnet server on the switch.
`Telnet / Start Session`	Starts a Telnet session to use the CLI. After starting a Telnet session you will need to login to the switch to use the CLI.
`SSH / Status`	Displays the current SSH status, either '`enabled`' or '`disabled`'.
`SSH / Configure Settings`	Configures the SSH Status. Select '`enabled`' or '`disabled`' to configure the status of the SSH server on the switch. Note that relevant certificates must be installed to initiate an SSH session.
`SSH / Start Session`	Starts a secure SSH session to use the CLI. After starting an SSH session you will need to login to the switch to use the CLI.

Description Active Sessions

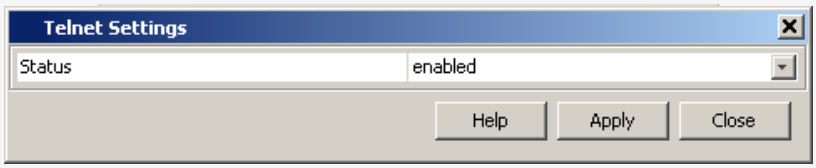
Label / Field / Button	Description
`Active Sessions / Connection Type`	A Console connection or a VTY connection.
`Active Sessions / Username`	Login name for a user.
`Active Sessions / Privilege Level`	The privilege set for a user for VTY or console connection. Privilege levels range from 0-15 with 15 the highest privilege level. Privilege levels are used in the CLI to enable or disable access to different configuration modes and commands. Privilege levels 0-14 only enables users to view system configuration and system behavior. Privilege level 15 enables users to globally configure all the interfaces on a switch.
`Active Sessions / Idle Time`	Time in seconds that the SSH Server waits to receive data from the SSH Client. The SSH Server disconnects when the Idle Time limit is reached.
`Active Sessions / IP Address`	The IPv4 address for the VTY connection.

Management > Remote CLI Access > Telnet Settings

The **Management** > **Remote CLI Access** > **Telnet Settings** dialog allows you to enable or disable Telnet.

Configuration Dialog

Example showing **Management** > **Remote CLI Access** > **Telnet Settings** dialog:



Description

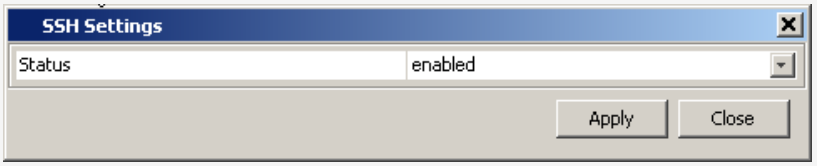
Label / Field / Button	Description
`Status`	Select `enabled` or `disabled` from the drop down list on this dialog to enable or disable Telnet respectively on the switch.

Management > Remote CLI Access > SSH Settings

The **Management** > **Remote CLI Access** > **SSH Settings** dialog allows you to enable or disable SSH.

Configuration Dialog

Example showing the **Management** > **Remote CLI Access** > **SSH Settings** dialog:



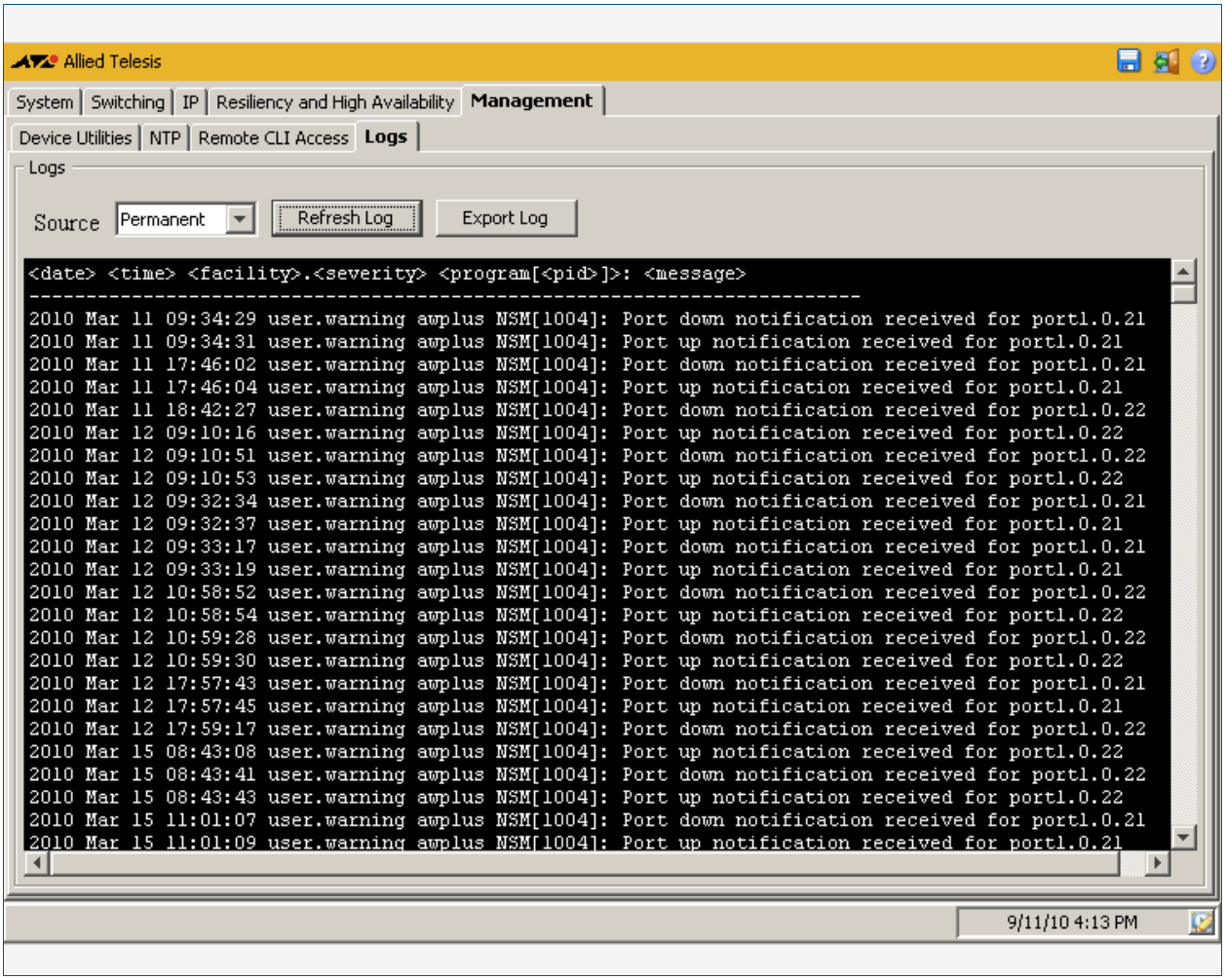
Description

Label / Field / Button	Description
`Status`	Select `enabled` or `disabled` from the drop down list on this dialog to enable or disable SSH respectively on the switch.

Management > Logs

The **Management** > **Logs** menu tab allows you to view the switch logs, and export the switch logs as .csv format files.

Menu Tab Example showing the **Management** > **Logs** menu tab:



Description

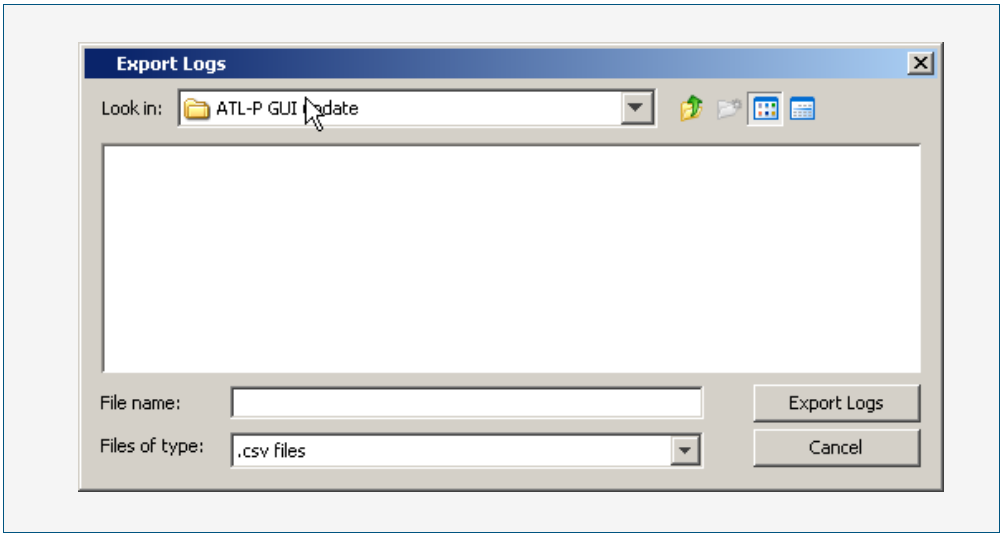
Label / Field / Button	Description
`Logs`	Display, select and export the available switch log files for troubleshooting use.
`Source`	Select the buffered log or the permanent log available on the switch to display or export to a .csv format file for use in a spreadsheet.
`Source / Refresh Log`	Select this button to display an updated buffered or permanent log.
`Source / Export Log`	Select this option to export the log to a .csv format file for use in a spreadsheet.

Management > Logs > Export Logs

The **Management** > **Logs** > **Export Logs** dialog allows you to export the switch logs as .csv format files.

Configuration Dialog

Example showing the **Management** > **Logs** > **Export Logs** dialog:



Description

Label / Field / Button	Description
`File name:`	Enter the file name for the exported log file.
`Files of type:`	Select `.csv files` to export the log file as a comma separated file, so each column of the log file can be formatted in a spreadsheet.

Appendix D: Glossary

Numerics D.2
A D.2
B D.5
C D.7
D D.9
E D.12
F D.13
G D.14
H D.14
I D.14
L D.16
M D.18
N D.20
O D.21
P D.21
Q D.24
R D.24
S D.27
T D.30
U D.31
V D.32
W D.33

Numerics

10BaseT

10 Mbps/baseband/twisted pair. The IEEE standard for twisted pair Ethernet.

802.1X

IEEE 802.1x is an IEEE Standard for port-based Network Access Control (**NAC**). It is part of the IEEE 802.1 group of networking protocols. It provides an authentication mechanism to devices wishing to attach to a LAN, either establishing a point-to-point connection or preventing it if authentication fails. It is used for securing wireless 802.11 access points and is based on the Extensible Authentication Protocol (**EAP**). Authentication is required on a per-port basis. The main components of an 802.1X implementation are:

- The authenticator - the port on this device that wishes to enforce authentication before allowing access to services that are accessible behind it.
- The supplicant - the port that wishes to access services offered by the authenticator's system. The supplicant may be a port on a PC or other device connected to this device.
- The authentication server - a device that uses the authentication credentials supplied by the supplicant, via the authenticator, to determine if the authenticator should grant access to its services.

See **AAA** and **Tri-authentication**.

For a configuration example see **"Configuring 802.1X" on page 68.7**. For a sample configuration script see **"Sample 802.1X-Authentication Configuration" on page 66.7**.

A

AAA

AAA is the collective title for the three related functions of Authentication, Authorization and Accounting. These function can be applied in a variety of methods with a variety of servers.

Authentication is performed in the following contexts:

- Login authentication of user shell sessions on the console port, and via telnet/SSH.
- **802.1X** authentication of devices connecting to switch ports.
- **MAC authentication** of devices connecting to switch ports.
- **Web-authentication** of devices connecting to switch ports.

Accounting is performed in the following contexts:

- Accounting of console login sessions.
- Accounting of 802.1x authenticated connections.
- Accounting of MAC authenticated connections.
- Accounting of Web authenticated connections.

There are two types of servers that can be used:

- Local user database.
- **RADIUS** servers.

When 802.1X authentication, MAC authentication and Web-authentication are configured to run simultaneously on a switch port this is called tri-authentication.

For more information see **Chapter 66, AAA Introduction and Configuration**. For a configuration example see **"Configuring AAA Login Authentication" on page 66.5**. For sample 802.1x, MAC authentication and Web-authentication configuration scripts see **"Sample Authentication Configurations" on page 66.7**.

Access-list

See **ACL**.

ACL

Access Control List. An ACL is one filter, or a sequence of filters, that are applied to an interface to either block, pass, or when using QoS, apply priority to, packets that match the filter definitions. ACLs are used to restrict network access by hosts and devices and to limit network traffic. See **ACL sequence numbers** and **ACL types**.

For more information see **Chapter 57, Access Control Lists Introduction**.

ACL sequence numbers

To help manage **ACLs** you can apply sequence numbers to filters. This allows you to remove filters from named and numbered ACLs without having to reconfigure an ACL. The ability to add sequence numbers to filters simplifies updates through the ability to position a filter within an ACL. When you add a new filter, you can specify a sequence number to position the filter in the ACL and you can also remove a current filter in an ACL by specifying a sequence number.

For more information see **"ACL Filter Sequence Numbers" on page 57.14**.

ACL types

ACLs are separated into two different types, software ACLs and hardware ACLs.

Hardware ACLs are applied directly to an interface, or are used for QoS **Classifications**. They can be either named, or can use the following numeric ranges:

- 3000-3699 for Hardware IP ACLs
- 4000-4699 for Hardware MAC ACLs

For more information see **"Defining Hardware IP ACLs" on page 57.6** and **"Defining Hardware MAC ACLs" on page 57.5**.

Software ACLs can be either named ACLs, using the standard or extended keyword followed by a text string, or they can use the following numeric ranges:

- 1-99
- 100-199
- 1300-1999
- 2000-2699

Software ACLs are used in features such as SNMP, IGMP, BGP and OSPF.

Active master

The switch that manages the stack, or **VCStack**, also referred to as the **Stack master**.

See **Disabled master** for information about how this relates to **Stack master** or **Active master**.

Address resolution

The process of resolving and mapping hardware MAC addresses into their corresponding network layer IP addresses. Depending on the underlying network, address resolution may require broadcasts on a local network.

For more information see **"ARP" on page D.4**.

Adjacency

A state existing between two OSPF routers. These routers build their routing databases by exchanging link state advertisements, often termed hello messages. When a pair has completed the process, the routers are said to be "adjacent."

AMF (also shown as ATMF in commands and output)

AMF (Allied Telesis Management Framework) is a suite of features that combine to simplify network management across all supported network switches from the core to the edge. Its major features are:

- Unified Command Line
- Configuration Backup and Recovery
- Rolling-Reboot Upgrade
- Node Provisioning

For more information see **"AMF Introduction and Configuration" on page 85.1**.

ARP

Address Resolution Protocol. ARP is used by your device to dynamically learn the Layer 2 address of devices in its networks. Most hosts also have a MAC physical address in addition to the assigned IP address. For Ethernet, this is a 6-byte, globally unique number. ARP enables your device to learn the physical address of the host that has a given IP address.

For more information see **"Address Resolution Protocol (ARP)" on page 28.3**.

AS

Autonomous System. A group of networks with a common routing infrastructure. An AS runs interior gateway protocols (IGPs) such as **RIP** and **OSPF** within its boundaries and uses exterior gateway protocols (EGPs) such as **BGP** to exchange routing information with other ASs.

ASCII

The *American Standard Code for Information Interchange*. A standard character-to-number encoding widely used within the computer industry.

ASIC

Application Specific Integrated Circuit. An integrated circuit (chip) manufactured to perform a specific function.

Asynchronous

Transmission in which each character is sent individually. The time intervals between transmitted characters may be of unequal length. Transmission is controlled by start and stop elements before and after each character. See **"Synchronous" on page D.30**

Autonegotiation

Autonegotiation lets the port adjust its speed and duplex mode to accommodate the device connected to it. When the port connects to another autonegotiating device, they negotiate the highest possible speed and **Duplex mode** for both of them.

Autonomous system

See **AS**.

B

BGP

Border Gateway Protocol. BGP is an exterior gateway protocol that determines the best path in networks, performs optimal routing between multiple autonomous systems or domains, and exchanges routing information with other BGP systems. The RFCs 1771 (BGP4), 1654 (first BGP4 specification), and 1105, 1163, 1267 (older version of BGP) describe BGP and BGP4.

BGP filter types

There are four filter types that can be applied to the BGP updates being exchanged between BGP peers:

- Distribute filters - these use ACLs and look at the individual prefixes within an update message. If a prefix within the update message matches the filter criteria then that individual prefix is filtered out or accepted depending on what action the filter entry has been configured to carry out. Note that you cannot combine distribute filters and prefix filters.
- Path filters - these look at the AS-Path attribute in update messages. If the AS-Path attribute in the update matches the filter criteria then the whole update message is filtered out or accepted, depending on what action the filter entry has been configured to carry out.
- Prefix filters - these use prefix lists and look at the individual prefixes within an update message. If a prefix within the update message matches the filter criteria then that individual prefix is filtered out or accepted depending on what action the filter entry has been configured to carry out. Note that you cannot combine distribute filters and prefix filters.
- Route maps - these have a complex combination of match criteria and actions. They can be used to filter out routes and also to alter the attributes in update messages.

All these filter types can be used in incoming or outgoing directions. Therefore, all the filters can all be used to filter the update packets that are received from a peer, or the update packets which the router itself is sending to a peer.

BGP peers

Within the BGP protocol, the exchange of routing information is carried out between pairs of routers. Two routers create a TCP connection with each other, and exchange routing information as specific data packets within that TCP session. The routers at the ends of the TCP connection are referred to as BGP peers. Any given router can form peering relationships with multiple routers.

BGP updates

Once a router has established a BGP connection with a peer, it will start to exchange routing information with that peer. A BGP update message is the packet that is used to transfer the routing information.

BIST

Built In Self Test. A mechanism that permits the device to test itself.

Blackhole route

A blackhole route is a routing table entry that does not forward packets. A blackhole route is specified as an interface with an **ip route** command. Note that a blackhole route is also called a **Null route**.

B-MAC

Backbone MAC address.

BPDU

Bridge Protocol Data Unit. A **Spanning tree** protocol initializing packet sent at configurable intervals to exchange information among bridges in the LAN.

For information on the standardized format for MSTP BPDU messages see **"MSTP Bridge Protocol Data Units (BPDUs)" on page 20.17**.

Bridge

A device that connects two or more networks and forwards packets between them. Bridges function at the data link layer or Layer 2 of the OSI reference model. A bridge will filter, send or flood an incoming frame, base on the MAC address of that frame.

Broadcast

One device sends out data that is intended to be received and processed by every device that it reaches.

Broadcast domain

A section of an Ethernet network comprising all the devices that will receive broadcast packets sent by any device in the domain. Separated from the rest of the network by a Layer 3 switch.

BOOTP

Bootstrap Protocol. BOOTP is a UDP-based protocol that enables a booting host to dynamically configure itself without external interventions. A BOOTP server responds to requests from BOOTP clients for configuration information, such as the IP address the client should use.

B-TAG

Backbone TAG Field.

B-VID

Backbone VLAN ID (tunnel).

B-VLAN

Backbone VLAN (tunnel).

C

CHAP

Challenge Handshake Authentication Protocol. CHAP is an authentication method used by PPP servers to validate the identity of clients. CHAP verifies the identity of the client by using a three-way handshake, and the verification is based on a shared secret by the client and the server, such as the client's password.

CIST

Common and Internal Spanning Tree. The CIST is the default spanning tree instance of **MSTP**, i.e. all VLANs that are not members of particular **MSTIs** are members of the CIST. Also, an individual MST region can be regarded as a single virtual bridge by other MST regions. The spanning tree that runs between regions is the CIST. The CIST is also the spanning tree that runs between MST regions and Single Spanning Tree (SST) entities.

For more information see **"Common and Internal Spanning Tree (CIST)" on page 20.15**.

Classification

In **ACLs** and **QoS**, classification is the process of filtering and marking. Filtering involves sorting your data into appropriate traffic types. Marking involves tagging the data so that downstream ports and routers can apply appropriate service policy rules. There are two reasons to classify data:

- To provide network security (security ACLs).
- To apply service quality criteria QoS.

The main application of security ACLs is to block undesired traffic. When using ACLs though QoS, the same classification and action abilities are available, but QoS has some additional fields that it can match on and also provides the ability to perform metering, marking and remarking on packets that match the filter definitions.

For more information on QoS classification see **"Classifying Your Data" on page 62.7**.

Class maps

Class maps are among the pivotal **QoS** components. They provide the means that associate the classified traffic with its appropriate QoS actions. They are the linking elements for the following functions:

- **Classification**.
- policy mapping. See **Policy maps**.
- **Premarking**.

The relationship between a class map and a policy map can be one-to-one or many-to-one.

For more information see **"Class Maps" on page 62.7**.

CLI

Command Line Interface. With three distinct modes, the CLI is very secure. In User exec mode you can view settings and troubleshoot problems but you cannot make changes to the system. In Privileged exec mode you can change system settings and restart the device. You can only make configuration changes in Global configuration mode, which reduces the risk of making accidental configuration changes.

For more information see **"How to Work with Command Modes" on page 1.7**.

C-MAC

Customer MAC Address.

Collision domain

A physical region of a local area network (LAN) in which data collisions can occur.

Continuous reboot prevention

The continuous reboot prevention feature allows the user to configure a switch to stop rebooting if the device gets into a cycle of continuous rebooting. The user can configure the time period, the maximum number of times the switch can reboot within the specified time period, referred to as the threshold, and the action to take if the threshold is exceeded.

For more information see **"Continuous Reboot Prevention" on page 1.35**.

Control VLAN

In **EPSR**, the VLAN over which all control messages are sent and received. EPSR never blocks this VLAN.

For more information see **"Ring Components and Operation" on page 83.2**.

CoS

Class of Service. CoS is a method for classifying traffic on a packet by packet basis using information in the type-of-service (ToS) byte to provide different service levels to different traffic. See **QoS**.

For more information see **"CoS to Egress Queue Premarking" on page 62.11**.

Cost

An indication of the overhead required to send packets across a certain interface.

C-TAG

Customer VLAN TAG.

C-VID

Customer VLAN ID.

C-VLAN

Customer VLAN.

D

Data VLAN

In **EPSR**, a VLAN that needs to be protected from loops. Each EPSR domain has one or more data VLANs.

For more information see **"Ring Components and Operation" on page 83.2**.

Designated bridge

Each bridge or LAN in the **Spanning tree**, except the **Root bridge**, has a unique parent, known as the designated bridge. Each LAN has a single bridge, called the designated bridge, that connects it to the next LAN on the path towards the root bridge.

For an overview of spanning tree operation see **"Spanning Tree Operation" on page 20.2**.

DHCP

Dynamic Host Configuration Protocol. A method of automatically allocating IP addresses. A DHCP server holds a pool of IP addresses from which it draws individual ones as it allocates them to users when they log on.

For more information see **Chapter 89, Dynamic Host Configuration Protocol (DHCP) Introduction**.

DHCP Leasequery

The DHCP Leasequery protocol (RFC 4388) allows a device or process, for example a DHCP relay agent, to obtain IP address information directly from the DHCP server using DHCPLEASEQUERY messages.

For more information see **"Enable DHCP Leasequery" on page 89.5**.

DHCP lease probing

Probing is used by the DHCP server to check whether an IP address it wants to lease to a client is already being used by another host. Probing is configured on a per-DHCP pool basis.

For more information see **"DHCP Lease Probing" on page 89.8**.

DHCP relay agent option 82

Enabling the DCHP Relay Agent Option 82 feature on the switch allows the switch to insert extra information into the DHCP packets that it is relaying. The information is stored in a specific optional field in the DHCP packet, namely, the agent-information field, which has option ID 82.

Note that the DHCP Relay Agent Option 82 agent information inserted by the DHCP snooping differs from the information added by DHCP Relay. The switch cannot be configured to use both the DHCP relay agent option and DHCP snooping.

For information about the DHCP Relay Agent Option 82 information inserted by DHCP Relay see **"DHCP Relay Agent Information Option (Option 82)" on page 89.10**.

For information about the DHCP Relay Agent Option 82 information inserted by DHCP snooping see **"DHCP Relay Agent Option 82" on page 79.4**.

DHCP relay agents

DHCP relay agents pass BOOTP and DHCP messages between servers and clients. Networks where the DHCP or BOOTP server does not reside on the same IP subnet as its clients need the intermediate routers to act as relay agents.

For information on how to configure the DHCP relay agent see **"DHCP Relay Agent Introduction" on page 89.9**.

DHCP snooping

DHCP snooping provides an extra layer of security on the switch via dynamic IP source filtering. DHCP snooping filters out traffic received from unknown, or 'untrusted' ports, and builds and maintains a DHCP snooping database.

With DHCP snooping, IP sources are dynamically verified, and filtered accordingly. IP packets that are not sourced from recognized IP addresses can be filtered out. This ensures the required traceability.

For more information see **Chapter 79, DHCP Snooping Introduction and Configuration**. For a configuration example see **"Configure DHCP Snooping" on page 79.10**.

Digital Diagnostics Monitoring (DDM)

Modern optical SFP transceivers support Digital Diagnostics Monitoring (DDM) functions. This feature allows you to monitor real-time parameters of the SFP, such as optical output power, optical input power, temperature, laser bias current, and transceiver supply voltage. Additionally, RX LOS (Loss of Signal) is shown when the received optical level is below a preset threshold. Monitoring these parameters allows you to check on the health of all installed SFPs or a specific SFP transceiver. See also **Trouble-shoot Fiber and Pluggable Issues** in **Getting Started**.

SFP transceivers that support DDM display the following output from the **show system pluggable detail** and the **show system pluggable diagnostics** commands for monitoring SFPs:

- **Diagnostic Calibration**: Specifies whether the pluggable supports DDM Internal Calibration. **Internal** is displayed if the pluggable supports DDM Internal Calibration.
- **Power Monitoring**: Displays the received power measurement type, which can be either **OMA** (Optical Module Amplitude) or **Avg** (Average Power) measured in µW.

For further information about the DDM feature on the switch for installed SFP transceivers, see the **show system pluggable detail** command and the **show system pluggable diagnostics** command in **Chapter 10, System Configuration and Monitoring Commands**.

Disabled master

The Disabled Master is a variant of the **Stack master** or **Active master** and is used with the **DMM (disabled master monitoring)**feature. The Disabled Master has the same configuration as the **Stack master** or **Active master**, but has all its switchports disabled. The Disabled Master is only used is the stack separates into two stubs. By having all switchports disabled, the Disabled Master avoids potentially detrimental network connectivity problems from having two Stack Masters or Active Masters having the same configuration. The Stack Master's or Active Master's ports are unaffected by the Disabled Master's ports, so the Stack Master or Active Master continues to forward traffic normally.

For information about the Disabled Master and the Disabled Master Monitoring feature, see the **Disabled Master Monitoring (DMM)** section in **Chapter 110, VCStack Introduction** and the **stack disabled-master-monitoring** command in **Chapter 111, Stacking Commands**.

DLF

Destination Lookup Failure. DLF is the event of receiving a unicast Ethernet frame with an unknown destination address.

DMM (disabled master monitoring)

The Disabled Master Monitoring (DMM) features checks the status of the Active Master via the Stack Resiliency Link. If the Active Master fails then the Disabled Master changes state to Active Master. A Disabled Master has the same configuration as the Active Master, but has all links shutdown. This change in state for the Disabled Master to become the Active Master allows traffic forwarding to continue on the VCStack.

For information about the Disabled Master and the Disabled Master Monitoring feature, see the **Disabled Master Monitoring (DMM)** section in **Chapter 110, VCStack Introduction** and the **stack disabled-master-monitoring**

DNS

Domain Name System. DNS allows you to access remote systems by entering human-readable device host names rather than IP addresses. DNS works by creating a mapping between a device name, such as **www.alliedtelesis.com**, and its IP address. These mappings are held on DNS servers. The benefits of DNS are that domain names:

- Can map to a new IP address if the host's IP address changes.
- Are easier to remember than an IP address.
- Allow organizations to use a domain name hierarchy that is independent of any IP address assignment.

For more information see **"Domain Name System (DNS)" on page 28.8**.

DNS relay

DNS Relay provides the presence of a local virtual DNS server on your AlliedWare Plus™ device which can service DNS lookup requests sent to it from local hosts. The DNS Relay will usually relay the requests to an external, or upstream, DNS server.

For more information see **"DNS Relay" on page 28.11**.

DoS

Denial of Service. A generic term for attacks that reduce or stop the operation of a network.

DSCP value

The Differentiated Services Code Point within the TOS field of an IP packet header. This is a 6-bit number in the range 0-63.

Duplex mode

See **Full duplex** and **Half duplex**.

Dynamic channel group

A dynamic channel group also known as a LACP channel group, an etherchannel, or a LACP aggregator, enables a number of ports to be dynamically combined to form a single higher bandwidth logical connection. See **LACP**.

For an more information see **"Link Aggregation Control Protocol (LACP)" on page 22.3**. For a configuration example see **"Configuring an LACP Channel Group" on page 22.5**.

Dynamic Link Failover

Dynamic Link Failover (Host Attach) is a versatile feature that enables devices that do not support link aggregation to form multiple active links by using **Triggers** and **Scripts**. You can customize Dynamic Link Failover to suit almost any situation, from a simple redundant backup link to multiple active links capable of basic load-sharing.

E

EAP

Extensible Authentication Protocol. EAP carries out the authentication exchange between the supplicant and the authentication server.

EEE

The IEEE 802.3az Energy Efficient Ethernet (EEE) standard is a specification for lowering the power consumption of Ethernet devices during periods of low link utilization. If no data is being sent then the Ethernet device can enter a sleep state, called Low Power Idle (LPI), to conserve the power consumed by the Ethernet device. See the **LPI** glossary entry.

Etherchannel

See **Dynamic channel group**.

Ethernet Protection Switching Ring

See **EPSR**.

EPSR

EPSR (Ethernet Protection Switching Ring) operates on physical rings of switches (note, not on meshed networks). When all nodes and links in the ring are up, EPSR prevents a loop by blocking data transmission across one port. When a node or link fails, EPSR detects the failure rapidly and responds by unblocking the blocked port so that data can flow around the ring. The EPSR components are:

- **EPSR domain**
- **Master node**
- **Transit node**
- **Ring port**
- **Primary port**
- **Secondary port**
- **Control VLAN**
- **Data VLAN**

For more information and example configurations see **Chapter 83, EPSR Introduction and Configuration**.

EPSR domain

A protection scheme for an Ethernet ring that consists of one or more data VLANs and a control VLAN.

For more information see **"Ring Components and Operation" on page 83.2**.

EGP

Exterior Gateway Protocol. EGP is an obsolete protocol that has been replaced by **BGP**. Not to be confused with the general term exterior gateway protocol.

Egress

Outgoing packet process.

Exterior Gateway Protocol

A protocol that distributes routing information to devices that connect separate autonomous systems (**ASs**).

F

FDB

Forwarding Database.

FIB

Forwarding Information Base. The **RIB** (Routing Information Base) populates the FIB with the best route to each destination. When your device receives an IP packet, and no filters are active that would exclude the packet, it uses the FIB to find the most specific route to the destination. If your device does not find a direct route to the destination, and no default route exists, it discards the packet and sends an ICMP message to that effect back to the source.

For more information see **"RIB and FIB Routing Tables" on page 35.4**.

Full duplex

When a port is in full duplex mode, the port transmits and receives data simultaneously. See **Half duplex**.

G

Guest VLAN

If **802.1X** authentication has been configured on access ports in the network, you might still want to provide limited network access to those users whose devices do not have 802.1x supplicant enabled, or who have unrecognized authentication credentials. The mechanism to achieve this is known as a Guest VLAN. The idea is that if the users device fails 802.1X authentication, or is not even performing any 802.1X authentication, then its connection port can be put into the guest VLAN.

For more information see **"Configuring a Guest VLAN" on page 2** and the **auth guest-vlan** command on page 65.9. For a configuration example see **"Configuring a Guest VLAN" on page 64.2**.

H

Half duplex

When a port is in half duplex mode, the port transmits or receives but not both at the same time. See **Full duplex**.

Hardware ACLs

See **ACL types**.

I

ICMP

Internet Control Message Protocol. ICMP allows networking devices to send information and control messages to other devices or hosts.

For more information see **"Internet Control Message Protocol (ICMP)" on page 28.13**.

ICMPv6

Internet Control Message Protocol Version 6. ICMPv6 is an implementation of **ICMP** for IPv6.

For more information see **"The Internet Control Message Protocol (ICMPv6)" on page 30.7**.

IGMP

Internet Group Management Protocol. IGMP is a communications protocol that hosts use to indicate that they are interested in receiving a particular multicast stream.

IGMP querier or router

A device in a subnetwork that is the coordinator for all multicast streams and IGMP membership information. Each subnet only has one active querier.

IGMP snooper

A device that spies on IGMP messages to create flow efficiencies by ensuring that multicast data streams are only sent to interested ports. A snooper can decide on the best path to send multicast packets at Layer 2 but does not initiate any IGMP communications.

For a configuration example see **"IGMP Snooping and Querier Configuration Example" on page 48.6**.

IGP

Interior Gateway Protocol. A routing protocol used within an autonomous system (**AS**).

Ingress

Incoming packet process.

Interior Gateway Protocol

See **IGP**.

IP directed broadcast

An IP directed broadcast is an IP packet whose destination address is a broadcast address for some IP subnet, but originates from a node that is not itself part of that destination subnet. When a directed broadcast packet reaches a switch that is directly connected to its destination subnet, the packet is flooded as a broadcast on the destination subnet. IP directed broadcast is enabled and disabled per VLAN interface. When enabled a directed broadcast packet is forwarded to an enabled VLAN interface if received on another subnet.

IP Helper

The IP Helper feature allows the switch to receive UDP broadcasts on one subnet, and forward them as broadcasts or unicasts into another subnet, so a client can use an application which uses UDP broadcast (such as Net-BIOS) when the client and server are located in different subnets. The IP Helper feature forwards UDP broadcast network traffic to specific hosts on another subnet and/or to the broadcast address of another subnet. When the IP Helper feature is enabled on a VLAN interface, the UDP broadcast packets received on the interface are processed for forwarding out through another interface into another subnet.

I-SID

Extended Service ID.

ISP

Internet Service Provider. An organization that offers its customers access to the Internet. The ISP connects its customers using a data transmission technology, such as dial-up or DSL.

I-TAG

Extended Service TAG.

L

LACP

Link Aggregation Control Protocol. LACP allows bundling of several physical ports to form a single logical channel providing enhanced performance and redundancy. The aggregated channel is viewed as a single link to each switch. The spanning tree views the channel as one interface and not as multiple interfaces. When there is a failure in one physical port, the other ports stay up and there is no disruption. LACP does not interoperate with devices that use Port Aggregation Protocol (PAgP).

For an more information see **"Link Aggregation Control Protocol (LACP)" on page 22.3**.

LACP aggregator

See **Dynamic channel group**.

LACP channel group

See **Dynamic channel group**.

LAG

See **Link aggregation group**.

Layer 3 switch

A Layer 3 switch is an optimized combination of routing software and specialized hardware. The software uses traditional methods (static routing commands, and routing protocols) to build up a table of the best routes to network destinations, and then writes them into a set of registers in the specialized forwarding hardware. The hardware then forwards packets, based on their Layer 3 address content, at very high data rates, using the values that are written into the registers.

LED

Light Emitting Diode (LED). An LED is a semiconductor that emits light by converting electrical energy. Power lights on switches and status lights on switch ports are LEDs. You can save power used by switch LEDs with the **ecofriendly led** command on page 10.13.

For more information and configuration examples see the section **"Save Power with the Eco-Friendly Feature" on page 1.32**. For command information and examples see the **ecofriendly led** command on page 10.13.

See also the **LPI** glossary entry.

Link aggregation group

A Link Aggregation Group is a collection of bundled switch ports for an aggregated link. Link aggregation is the bonding together of two or more data channels into a single channel that appears as single logical link of higher bandwidth increasing link performance and reliability.

For an more information see **"Link Aggregation Control Protocol (LACP)" on page 22.3**. For a configuration example see **"Configuring an LACP Channel Group" on page 22.5**

Link-local addresses

A link-local address is an IP (Internet Protocol) address that is only used for communications in the local network, or for a point-to-point connection. Routing does not forward packets with link-local addresses. IPv6 requires a link-local address is assigned to each interface, which has the IPv6 protocol enabled, and when addresses are assigned to interfaces for routing IPv6 packets.

Note that link-local addresses are retained in the system until they are negated by using the no variant of the command that established them. See the **ipv6 enable** command for more information.

Also note that the link-local address is retained in the system if the global address is removed using another command, which was not used to establish the link-local address. For example, if a link local address is established with the **ipv6 enable** command then it will not be removed using a **no ipv6 address** command.

LLDP

Link Layer Discovery Protocol. LLDP is a Layer 2 protocol that enables Ethernet network devices, such as switches and routers, to transmit and/or receive device-related information to or from directly connected devices on the network, and to store such information learned about other devices. LLDP is a link level ("one hop") protocol; LLDP information can only be sent to and received from devices that are directly connected to each other, or connected via a hub or repeater. Advertised information is not forwarded on to other devices on the network.

For more information see **Chapter 96, LLDP, LLDP-MED and Voice VLAN Introduction and Configuration**.

For configuration examples see **"Configuring LLDP" on page 96.11**.

LLDPDU

LLDP Data Unit. See **LLDP advertisements**.

LLDP advertisements

LLDP transmits advertisements as packets called LLDP Data Units (LLDPDUs). An LLDPDU consists of a set of Type-Length-Value elements (**TLV**), each of which contains a particular type of information about the device or port transmitting it.

LLDP-MED

Link Layer Discovery Protocol Media Endpoint Discovery. LLDP-MED is an enhancement to IEEE's 802.1AB LLDP, adding media and IP telephony-specific messages that can be exchanged between the network and endpoint devices.

For more information see **"LLDP-MED" on page 96.3**, **"LLDP-MED: Location Identification TLV" on page 96.7** and **"LLDP-MED Operation" on page 96.9**. For the procedure to configure LLDP-MED see **"Configure LLDP-MED" on page 96.14**.

Local RADIUS server

The local RADIUS Server enables you to create a RADIUS server on the switch.

For more information and configuration examples see **Chapter 72, Local RADIUS Server Introduction and Configuration**.

LPI

Low Power Idle (LPI). LPI is a feature of the IEEE 802.3az Energy Efficient Ethernet (EEE) standard. LPI lowers power consumption of switch ports during periods of low link utilization when connected to IEEE 802.3az compliant host devices. If no data is sent then the switch port can enter a sleep state, called Low Power Idle (LPI), to conserve power used by the switch.

For more information and configuration examples see the section **Save Power with the Eco-Friendly Feature** command on page 1.32. For command information and examples see the **ecofriendly lpi** command on page 10.14. See also the **LED** glossary entry.

LSA

Link State Advertisement. OSPF sends link-state advertisements (LSAs) to all other routers within the same hierarchical area. Data on attached interfaces, metrics used, and other variables, are included in OSPF LSAs. As OSPF routers accumulate link-state data, they use the Shortest Path First (SPF) algorithm to calculate the shortest path to each node.

M

MAC address learning

A key optimization in Ethernet switching is that the flooding of unicast traffic is minimized. This is based on switches knowing which port to forward traffic to for given destination MAC addresses. Switches achieve this by the simple process of noting on which ports packets arrive from given MAC addresses, as those will be the ports to which return packets to those MAC addresses will need to be forwarded. This process is referred to as MAC address learning.

MAC authentication

The way that MAC-based authentication works is that when the supplicant device starts sending packets, the authenticating switch will extract the source MAC address from the packets, and send a RADIUS request that uses this MAC address as the username and password in the request. See **AAA** and **Tri-authentication**.

For a sample configuration script see **"Sample MAC-Authentication Configuration" on page 66.8**.

Master node

In **EPSR**, the controlling node for a domain, responsible for polling the ring state, collecting error messages, and controlling the flow of traffic in the domain.

Master node states are:

- Complete - the state when there are no link or node failures on the ring.
- Failed - the state when there is a link or node failure on the ring. This state indicates that the master node received a Link-Down message or that the failover timer expired before the master node's secondary port received a Health message.

For more information see **"Ring Components and Operation" on page 83.2**.

MD5

Message Digest 5 authentication algorithm.

Metering

See **Policing**.

Metric

The sum of all the costs along the path to a given destination. See **Cost**.

MLD

Multicast Listener Discovery. MLD is used to exchange membership status information between IPv6 routers that support multicasting and members of multicast groups on a network segment. Host membership in a multicast group is reported by individual member hosts, and membership status is periodically polled by multicast routers.

MLD snooping

MLD snooping is a feature whereby a Layer 2 switch listens to or "snoops" the MLD messages passing through the switch or from member hosts and multicast routers. The purpose of MLD snooping is to provide efficient Layer 2 multicast forwarding, by sending only to hosts that have expressed an interest in receiving the multicast data.

For more information see **Chapter 56, MLD and MLD Snooping Introduction and Commands**.

MSTI

Multiple Spanning Tree Instance. **MSTP** enables the grouping and mapping of VLANs to different spanning tree instances. An MST Instance (MSTI) is a particular set of VLANs that are all using the same spanning tree.

For more information see **"Multiple Spanning Tree Instances (MSTI)" on page 20.12**.

MSTP

Multiple Spanning Tree Protocol. MSTP is similar to Rapid Spanning Tree Protocol (**RSTP**) - it provides loop resolution and rapid convergence. However it also has the extra advantage of making it possible to have different forwarding paths for different multiple spanning tree instances. This enables load balancing of network traffic across redundant links. A device running MSTP is compatible with other devices running RSTP or **STP**.

For more information see **"Multiple Spanning Tree Protocol (MSTP)" on page 20.11**. For a configuration example see **"Configuring MSTP" on page 20.19**.

MSTP regions

An MST region is a set of interconnected switches that all have the same values for the following MST configuration identification elements:

- MST configuration name - the name of the MST region.
- Revision level - the revision number of configuration.
- Configuration Digest - the mapping of which VLANs are mapped to which MST instances.

Each of the MST instances created are identified by an **MSTI** number. This number is locally significant within the MST region. Therefore, an MSTI will not span across MST regions.

For more information see **"MSTP Regions" on page 20.13**.

Multicast

One device sends out data that is intended to be received and processed by a selected group of the devices it reaches.

N

NAC

Network Access Control. NAC provides unprecedented control over user access to the network in order to mitigate threats to network infrastructure. NAC uses **802.1X** port-based authentication with standards-compliant dynamic VLAN assignment, to assess a user's adherence to the network's security policies, and either grant authentication or offer remediation. NAC also supports alternatives to 802.1x port-based authentication, such as **Web-authentication** to enable guest access, and **MAC authentication** for end points that do not have an 802.1x supplicant. Furthermore, if multiple users share a port then multi-authentication can be used and a **Guest VLAN** can be configured to provide a catch-all for users without an 802.1x supplicant.

For more information see **Chapter 68, 802.1X Introduction and Configuration** and **Chapter 64, Authentication Introduction and Configuration**.

NAS

Network Access Server. A NAS is a single point of access to a remote resource. The client connects to the NAS. The NAS then connects to another resource asking whether the client's supplied credentials are valid. Based on that answer the NAS then allows or disallows access to the resource. The NAS contains no information about what resources clients can connect to or what client credentials are valid. The NAS sends the credentials the client supplied to a resource which then validates the client.

Next hop

IP routing involves forwarding packets from one router to the next, until they reach their destination. Routers do not need to know the full path to a packet's destination, they just need to know the next router to forward the packet on to. This 'next router' is referred to as the next hop of an IP route.

Nested VLAN

See **VLAN double tagging**.

NTP

Network Time Protocol. NTP is a protocol for synchronizing the time clocks on a collection of network devices using a distributed client/server mechanism.

For more information see **Chapter 87, NTP Introduction and Configuration**.

Null route

A null route is a routing table entry that does not forward packets. A null route is specified as an interface with an **ip route** command. Note that a null route is also called a **Blackhole route**.

O

OSPF

Open Shortest Path First. A link-state routing protocol, OSPF is an interior gateway protocol (IGP) that uses the Shortest Path First (SPF) Dijsktra algorithm.

For more information see **Chapter 41, OSPF Introduction and Configuration**. For configuration examples see **"Enabling OSPF on an Interface" on page 41.10**, **"Setting Priority" on page 41.12**, **"Configuring an Area Border Router" on page 41.14**, , **"OSPF Cost" on page 41.15**, **"Configuring Virtual Links" on page 41.18** and **"OSPF Multi-Area Loopback Configuration" on page 41.23**

P

PAP

Password Authentication Protocol. PAP is an authentication protocol that uses a password and is used by PPP to validate users before allowing them to access server resources. PAP transmits plain text ASCII passwords over the network so it is not secure.

PDs

Powered Devices. PDs are devices such as IP phones, wireless LAN Access Points, and network cameras. PDs receive power, in addition to data, over existing network infrastructure and cabling. See **PoE**.

PIM-DM

Protocol Independent Multicast - Dense Mode. PIM-DM is a data-driven multicast routing protocol, which builds source-based multicast distribution trees that operate on the Flood-and-Prune principle. It requires unicast-reachability information, but does not depend on a specific unicast routing protocol. PIM-DM is a significantly less complex protocol than PIM-SM. PIM-DM works on the principle that it is probable that any given multicast stream will have at least one downstream listener. PIM-DM is ideal where many hosts subscribe to receive multicast packets, so most of the PIM Routers receive and forward all multicast packets.

For more information see **Chapter 54, PIM-DM Introduction and Configuration**. For a configuration example see **"PIM-DM Configuration" on page 54.4**.

PIM-SM

Protocol Independent Multicast - Sparse Mode. PIM-SM provides efficient communication between members of sparsely distributed groups - the type of groups that are most common in wide-area internetworks. PIM-SM is designed on the principle that several hosts wishing to participate in a multicast conference does not justify flooding the entire internetwork with periodic multicast traffic. PIM-SM is designed to limit multicast traffic so that only those routers interested in receiving traffic for a particular group receive the traffic.

For more information see **Chapter 50, PIM-SM Introduction and Configuration**. For configuration examples see **"Static Rendezvous Point configuration" on page 50.7**, **"Dynamic Rendezvous Point Configuration" on page 50.9** and **"Bootstrap Router Configuration" on page 50.10**.

Ping

Ping tests the connectivity between two network devices to determine whether each network device can "see" the other device.

Ping-of-death attack

A type of attack on a computer that involves sending a malformed or otherwise malicious ping to a network device.

Ping polling

Ping polling is used to ensure that a device is still present, live, and contactable in the network by periodically sending a packet to an IP address and waiting for a response. Configurable actions can be performed if responses are no longer arriving.

For more information see **Chapter 104, Ping Polling Introduction and Configuration**. For how to configure ping polling see **"Configuring Ping Polling" on page 104.4**.

PoE

Power over Ethernet. PoE is a mechanism for supplying power to network devices over the same cabling used to carry network traffic. PoE supplies power to network devices called Powered Devices (**PDs**).

For more information see **Chapter 24, Power over Ethernet Introduction**. For configuration examples see **"PoE and PoE+ Configuration" on page 24.14**.

Policing

In **QoS**, once you have set-up your **Classification** and created your **Class maps**, you can start conditioning your traffic flows. One tool used for traffic conditioning is the policer (or meter). The principle of policing is to measure the data flow that matches the definitions for a particular class-map; then, by selecting appropriate data rates, allocate the flows into one of three categories, Red Yellow or Green. You then decide what action to apply to the Red, Yellow and Green data.

For more information see **"Policing (Metering) Your Data" on page 62.14**.

Policy maps

Policy maps are the means by which you apply your **Class maps** to physical switch ports. A policy map can be assigned to several ports, but a port cannot have more than one policy map assigned to it. See **QoS**.

For more information see **"Policy Maps" on page 62.10**.

Port bit map

An efficient method for the storage of a list of ports. Each port is represented by a single bit in a 32-bit or 64-bit value.

Port mirroring

Port mirroring enables traffic being received and transmitted on a switch port to be sent to another switch port, the mirror port, usually for the purposes of capturing the data with a protocol analyzer. The mirror port is the only switch port that does not belong to a VLAN, and therefore does not participate in any other switching. Before the mirror port can be set, it must be removed from all trunk groups and all VLANs except the default VLAN.

PPP

Point-to-Point Protocol. A data link protocol used to establish a direct connection between two networking nodes. PPP can provide connection authentication and transmission encryption. PPPoE (Point-to-Point Protocol over Ethernet) is used over broadband connections as is PPPoA (Point-to-Point Protocol over ATM) with DSL.

Premarking

In **QoS**, premarking relates to adding QoS markers to your incoming data traffic before it is metered. QoS markers can be applied at both the link layer (within the CoS field), and at the network layer (within the DSCP field). See **Policing**.

For more information see **"Premarking and Remarking Your Traffic" on page 62.11**.

Primary port

In **EPSR**, a ring port on the master node. This port determines the direction of the traffic flow, and is always operational.

For more information see **"Ring Components and Operation" on page 83.2**.

Provisioning

Stack member provisioning is the pre-configuration of **Stack member** position ready for insertion at a later time. Provisioning enables a network administrator to pre-configure vacant stack member capacity within a **VCStack**, ready to be hot-swapped in at a later time. Later, when the stack member switch is physically added, its configuration is automatically applied with the minimum network disruption.

For more information see **"Provisioning (Stack Members)" on page 110.19**.

Proxy ARP

Proxy ARP allows hosts that do not support routing (i.e. they have no knowledge of the network structure) to determine the physical addresses of hosts on other networks.

For more information see **"Proxy ARP" on page 28.4**.

PSE

Power Sourcing Equipment. A device that can source power, such as an Ethernet switch, is termed Power Sourcing Equipment. Power Sourcing Equipment can provide power, along with data, over existing LAN cabling to Powered Devices (**PDs**). See **PoE**.

PSU

Power Supply Unit.

Q

Query Solicitation

Query Solicitation minimizes the loss of multicast data after a topology change on networks that use **EPSR** or spanning tree (**STP**, **RSTP**, or **MSTP**) for loop protection. Without Query Solicitation, when the underlying link layer topology changes, multicast data flow can stop for up to several minutes, depending on which port goes down and how much of the IGMP query interval remained at the time of the topology change. Query Solicitation greatly reduces this disruption.

For more information see **"Query Solicitation" on page 48.9**.

QoS

Quality of Service. QoS enables you to both prioritize traffic and limit its available bandwidth. The concept of QoS is a departure from the original networking protocols, in which all traffic on the Internet or within a LAN had the same available bandwidth. Without QoS, all traffic types are equally likely to be dropped if a link becomes oversubscribed. This approach is now inadequate in many networks, because traffic levels have increased and networks often carry time-critical applications such as streams of real-time video data. QoS also enables service providers to easily supply different customers with different amounts of bandwidth. Configuring Quality of Service involves two separate stages:

- Classifying traffic into flows, according to a wide range of criteria. Classification is performed by the switch's **Class maps**.
- Acting on these traffic flows.

For more information see **Chapter 62, Quality of Service (QoS) Introduction**.

Quality of Service

See **QoS**.

R

RADIUS

Remote Authentication Dial-In User Service. RADIUS is a networking protocol that provides centralized **AAA** (Authentication Authorization and Accounting) management for clients to a network. RADIUS is a client/server protocol that runs in the application layer, using UDP (User Datagram Protocol) for data transport. RADIUS authenticates users before granting them access to network resources and can account for the usage of network resources.

For more information see **Chapter 70, RADIUS Introduction and Configuration**. For configuration examples see **"RADIUS Configuration Examples" on page 70.15**.

Redistribute

Advertise routes learned from one routing protocol into another routing protocol.

Remarking

In **QoS**, remarking relates to adding QoS markers to your incoming data traffic after it is metered. QoS markers can be applied at both the link layer (within the CoS field), and at the network layer (within the DSCP field). See **Policing**.

Remote network MONitoring

See **RMON**.

Resiliency link

In **VCStack**, an extra, out-of-band, data link between stack members. In the event of loss of communication across the stacking connection, the stack members can determine the status of other members via communication on the resiliency link. This assists the stack members in deciding the correct course of action when communication on the stack is lost.

For more information see **"Stack Resiliency Link" on page 110.13**.

RIB

Routing Information Base. The RIB records all the routes that your device has learned. Your device uses the RIB to advertise routes to its neighbor devices and to populate the **FIB** (Forwarding Information Base).

For more information see **"RIB and FIB Routing Tables" on page 35.4**.

Ring port

In **EPSR**, a port that connects the node to the ring. On the master node, each ring port is either the primary port or the secondary port. On transit nodes, ring ports do not have roles.

For more information see **"Ring Components and Operation" on page 83.2**.

RIP

Routing Information Protocol. A simple distance vector IPv4 routing protocol, RIP is an Interior Gateway Protocol (IGP) that uses hop counts as its metrics. Given a choice of routes, RIP uses the route that takes the lowest number of hops. If multiple routes have the same hop count, RIP chooses the first route it finds.The AlliedWare Plus™ RIP module supports RFCs 1058 and 1723; the RIPv2 module supports more fields in the RIP packets, and supports security authentication features.

For more information see **"RIP" on page 34.2**. For configuration examples see **"Enabling RIP" on page 37.2**, **"Specifying the RIP Version" on page 37.4**, **"RIPv2 Authentication (Single Key)" on page 37.6**, **"RIPv2 Text Authentication (Multiple Keys)" on page 37.8** and **"RIPv2 md5 authentication (Multiple Keys)" on page 37.12**.

RIPng

Routing Information Protocol next generation. RIPng is a simple distance vector IPv6 routing protocol. It determines the number of hops between the destination and your device, where one hop is one link. Given a choice of routes, RIPng uses the route that takes the lowest number of hops. If multiple routes have the same hop count, RIPng chooses the first route it finds. RIPng is an extension of RIPv2 to support IPv6.

For configuration examples see **"Enabling RIPng" on page 39.2** and **"Troubleshooting RIPng Adjacency" on page 39.5**.

RIPv6

See **RIPng**.

RMON

Remote Network MONitoring. RMON was developed by the IETF to support monitoring and protocol analysis of LANs with a focus on Layer 1 and 2 information in networks. RMON is an industry standard that provides the functionality in network analyzers. An RMON implementation operates in a client/server model. Monitoring devices (or 'probes') contain RMON agents that collect information and analyze packets. The probes are servers and the Network Management applications that communicate with them are clients.

For more information see **Chapter 99, RMON Introduction and Configuration**. For a configuration example see **"RMON Configuration Example" on page 99.3**.

Roaming Authentication

Roaming Authentication improves the usability of network security by enabling users to move within the network without requiring them to re-authenticate each time they move. If a supplicant (client device) moves from one wireless access point to another wireless access point, and the wireless access points are connected to different ports, then the switch (authenticator) recognizes that the supplicant has been authenticated and accepts the supplicant without requiring re-authentication.

For more information see **"Web-Authentication" on page 64.4**.

Root bridge

A single **Bridge** is selected to become the **Spanning tree's** unique root bridge. This is the device that advertises the lowest Bridge ID. Each bridge is uniquely identified by its Bridge ID, which comprises the bridge's root priority (a spanning tree parameter) followed by its MAC address.

For an overview of spanning tree operation see **"Spanning Tree Operation" on page 20.2**.

Root path cost

A **Spanning tree** property. Each port connecting a **Bridge** to a LAN has an associated cost, called the root path cost. This is the sum of the costs for each path between the particular bridge port and the **Root bridge**. The **Designated bridge** for a LAN is the one that advertises the lowest root path cost. If two bridges on the same LAN have the same lowest root path cost, then the switch with the lowest bridge ID becomes the designated bridge.

For an overview of spanning tree operation see **"Spanning Tree Operation" on page 20.2**.

Route-map

A mechanism for filtering IP routes and changing their attributes.

RSTP

Rapid Spanning Tree Protocol. RSTP is an evolution of the Spanning Tree Protocol (**STP**) which provides for faster spanning tree convergence after a topology change. A device running RSTP is compatible with other devices running STP.

For more information see **"Rapid Spanning Tree Protocol (RSTP)" on page 20.8**. For a configuration example see **"Configuring RSTP" on page 20.9**.

S

SCP

Secure Copy Protocol. SCP allows for secure file transfer to and from the switch, protecting your network from unwanted downloads and unauthorized file copying.

For more information see **"Copying with Secure Copy (SCP)" on page 6.17**.

Script

A script is a sequence of commands stored as a plaintext file on a file subsystem accessible to the device, such as Flash memory. Each **Trigger** may reference multiple scripts and any script may be used by any trigger. When an event activates a trigger, the trigger executes the scripts associated with it in sequence. One script is executed completely before the next script begins.

See **Dynamic Link Failover**.

Secondary port

In **EPSR**, a second ring port on the master node. This port remains active, but blocks all protected VLANs from operating unless the ring fails. Similar to the blocking port in an STP/RSTP instance.

For more information see **"Ring Components and Operation" on page 83.2**.

sFlow

sFlow®[1] is an industry standard technology for monitoring high speed switched networks. It provides the ability to monitor traffic in data networks containing switches and routers.

For more information see **Chapter 106, sFlow Introduction and Configuration**. For how to configure sFlow see **"Configuring sFlow on your Switch" on page 106.5**.

sFlow agent

A network employing sFlow typically comprises a number of network (sFlow) agents that accumulate sampled data and traffic counter information. The agents then forward this data to a collector. The collector then analyses the information supplied by its agents in order to compile and display statistical profiles of the network and its traffic. The sFlow feature on your switch provides the sFlow agent capability.

For more information see **"The sFlow Agent" on page 106.3**.

sFlow collector

The sFlow collector receives traffic samples and counter information from a number of sFlow agents. These samples are received as a series of UDP datagrams. From the data contained within these datagrams, the collector is able to provide statistical and or graphical information of network traffic.

For more information see **"The sFlow Collector" on page 106.5**.

1. **sFlow® is a registered trademark belonging to InMon Corp, San Francisco, CA.**

SFTP

SSH File Transfer Protocol. SFTP provides a secure way to copy files onto your device from a remote device.

For more information see **"Copying with SSH File Transfer Protocol (SFTP)" on page 6.17**.

Software ACLs

See **ACL types**.

Spanning tree

A loop free portion of a network topology. The network topology is dynamically pruned to provide only one path for any packet. See **STP**, **RSTP** and **MSTP**.

Spanning Tree Protocol Root Guard

See **STP root guard**.

SSH

Secure Shell. SSH is a network protocol that uses strong authentication and encryption for remote access across a nonsecure network. SSH provides sessions between a host running a SSH server and a machine with a SSH client.

For more information see **Chapter 76, Secure Shell (SSH) Introduction**. For how to configure a SSH server see **"Configuring the SSH Server" on page 76.4**. For how to configure a SSH client see **"Configuring the SSH Client" on page 76.9**.

Stack

See **VCStack**.

Stack master

The switch that manages the stack, or **VCStack**, also referred to as the **Active master**.

See **Disabled master** for information about how this relates to **Stack master** or **Active master**.

Stack member

An individual switch that is part of a **VCStack**.

S-TAG

Service VLAN TAG.

Static aggregator

See **Static channel group**.

Static channel group

A static channel group, also known as a static aggregator, enables a number of ports to be manually configured to form a single logical connection of higher bandwidth. By using static channel groups you increase channel reliability by distributing the data path over more than one physical link.

Storm-control

Storm-control enables you to specify the threshold level for broadcasting, multicast, or destination lookup failure (DLF) traffic for a port. Storm-control limits the specified traffic type to the specified threshold.

For more information see **"Storm Control" on page 16.12**.

Storm protection

Storm protection uses **QoS** mechanisms to classify on traffic likely to cause a packet storm (broadcast and multicast). With QoS storm protection, several actions are possible when a storm is detected:

- You can disable the port physically.
- You can disable the port logically.
- You can disable the port for a particular VLAN.

For more information see **"Storm Protection" on page 62.24**.

STP

Spanning Tree Protocol. STP is the original bridge protocol defined by IEEE standard 802.1D-1988. It creates a single spanning tree over a network.

For more information see **"Spanning Tree Protocol (STP)" on page 20.5**. For a configuration example see **"Configuring STP" on page 20.6**.

STP root guard

Spanning Tree Protocol Root Guard. STP Root Guard designates which devices can assume the role of **Root bridge** in an STP network. This stops an undesirable device from taking over this role, where it could either compromise network performance or cause a security weakness.

See the **spanning-tree guard root** command on page 21.42.

Subnet address

A subnet portion of an IP address. In a subnetted network, the host portion of an IP address is split into a subnet portion and a host portion using an address or subnet mask.

Subnet mask

A bit mask used to select bits from an Internet address for subnet addressing. The mask is 32 bits long and selects the network portion of the Internet address and one or more bits of the local portion. Sometimes called address mask.

Superloop

Within an EPSR ring configuration, a superloop is a data loop whose path traverses more than a single EPSR ring. This occurrence is a fault condition that is usually do to a break in a physical segment that is shared by the two rings. For a superloop condition to occur, the two physical rings must share one or more data VLANs. Superloops can be prevented by employing Superloop Protection. For more information, See **"Superloop Protection" on page 83.18**.

Switch instance

A single switch chip with its associated ports, internal data interfaces, hardware tables, and packet buffer memory.

S-VID

Service VLAN ID.

S-VLAN

Service VLAN.

Synchronous

Transmission in which the data characters and bits are transmitted at a fixed rate with the transmitter and receiver synchronized. This eliminates the need for start-stop elements, as in asynchronous transmission, but requires a flag character to be transmitted when there is no data to transmit. See **"Asynchronous" on page D.5**

T

TACACS+

TACACS+ (Terminal Access Controller Access-Control System Plus) provides a method for securely managing multiple network access points from a single management service. TACACS+ is a TCP-based access control protocol that allows a device to forward a user's username and password to an authentication server to determine whether access can be allowed. In addition to this authentication service, TACACS+ can also provide authorization and accounting services. One of the features of TACACS+ is the ability to separate authentication, authorization and accounting so that these functions can be provided independently on separate servers.

For information on the AlliedWare Plus implementation of TACACS+, see **Chapter 74, TACACS+ Introduction and Configuration** and **Chapter 75, TACACS+ Commands**.

TCN

Topology Change Notification.

Thrash limiting

MAC address thrashing occurs when MAC addresses move rapidly between one or more ports or trunks, for example, due to a network loop. Thrash limiting enables you to apply actions to a port when thrashing is detected. It is supported on all port types and also on aggregated ports.

For more information see **"Thrash Limiting" on page 16.14**

TLV

Type-Length-Value. A single **LLDPDU** contains multiple TLVs. TLVs are short information elements that communicate complex data, such as variable length strings, in a standardized format. Each TLV advertises a single type of information, such as its device ID, type, or management addresses. See **LLDP advertisements**.

Traceroute

Traceroute is used to discover the route that packets pass between two systems running the IP protocol. Traceroute sends an initial UDP packets with the Time To Live (TTL) field in the IP header set starting at 1. The TTL field is increased by one for every subsequent packet sent until the destination is reached. Each hop along the path between two systems responds with a TTL exceeded packet (ICMP type 11) and from this the path is determined.

Transit node

In **EPSR**, nodes other than the master node in the domain.

Transit node states are:

- Idle - the state when EPSR is first configured, before the master node determines that all links in the ring are up. In this state, both ports on the node are blocked for the data VLAN. From this state, the node can move to Links Up or Links Down.
- Links Up - the state when both the node's ring ports are up and forwarding. From this state, the node can move to Links Down.
- Links Down - the state when one or both of the node's ring ports are down. From this state, the node can move to Preforwarding.
- Pre-forwarding - the state when both ring ports are up, but one has only just come up and is still blocked to prevent loops. From this state, the transit node can move to Links Up if the master node blocks its secondary port, or to Links Down if another port goes down.

For more information see **"Ring Components and Operation" on page 83.2**.

Tri-authentication

Authentication commands enable you to specify three different types of device authentication: **802.1X** authentication, **MAC authentication**, and **Web-authentication**. All three types can be configured to run simultaneously on a switch port. The simultaneous configuration and authentication of all three types on a port is called tri-authentication.

For a configuration example see **"Tri-Authentication Configuration" on page 64.24**.

Trigger

A trigger is an ordered sequence of scripts that is executed when a certain event occurs. Each trigger may reference multiple scripts and any **Script** may be used by any trigger. When an event activates a trigger, the trigger executes the scripts associated with it in sequence. One script is executed completely before the next script begins.

See **Dynamic Link Failover**.

Type-Length-Value

See **TLV**.

U

Unicast

Two individual devices hold a conversation just between themselves.

V

VCStack

A group of two or more switches operating as a single switch. See **Virtual Chassis Stacking**.

VCStack fast failover

VCStack Fast Failover provides absolutely minimal network downtime in the event of a problem with the stack.

See the **reboot rolling** command on page 111.5.

VID

VLAN Identifier or VLAN ID. When you create a VLAN you give it a numerical VID which is included in VLAN-tagged Ethernet frames to and from this VLAN.

Virtual Chassis Stacking

Virtual Chassis Stacking (VCStack™) is the name given to two or more Allied Telesis switches that are configured to operate as a single switch. From a configuration and management point of view, it is as though the switches are just one device with a seamless transition from the ports of one stack member to the ports of the next.

For more information see **Chapter 110, VCStack Introduction**.

VLAN classification

A packet can be allocated VLAN membership based on its protocol, subnet, or port.

VLAN double tagging

VLAN double tagging is used operate a number of private Layer 2 networks within a single public Layer 2 network. A VLAN double tagging implementation consists of the following port types:

- Provider ports - these connect to a service provider's Layer 2 network
- Customer edge ports - these connect to a customer's private Layer 2 network

For more information see **"VLAN Double Tagging (VLAN Stacking)" on page 18.5**. For a configuration example see **"Configuring Double-Tagged VLANs" on page 18.6**.

VLAN ID

See **VID**.

VLAN identifier

See **VID**.

VLAN stacking

See **VLAN double tagging**.

VLAN tag

IEEE standard 802.1q defines an additional 4 byte tag field that can be inserted immediately following the MAC address, plus any routing fields present. This field contains a 12 bit VLAN identifier, commonly referred to as the VLAN tag. The VLAN tag is used to determine which VLAN a given frame should be forwarded to.

Other tags included in the 802.1q tag field is a Tag Protocol Identifier tag, and a Type of Service tag used to determine data priority.

Voice VLAN

Voice VLAN automatically separates voice and data traffic into two different VLANS. This automatic separation places delay-sensitive traffic into a voice-dedicated VLAN, which simplifies QoS configurations.

For more information see **"Voice VLAN" on page 96.3**.

VoIP

Voice over Internet Protocol. Enables the delivery of voice communications over IP networks such as the Internet or other packet-switched networks instead of over traditional telephony circuits.

VRID

Virtual Router Identifier.

VRRP

Virtual Router Redundancy Protocol. VRRP combines two or more physical switches into a logical grouping called a virtual router. The physical switches then operate together to provide a single logical gateway for hosts on the LAN. If the master fails, the other devices assume the virtual IP address.

For more information see **Chapter 81, VRRP Introduction and Configuration**. For configuration examples see **"VRRP Configuration Examples" on page 81.13**.

W

Web-authentication

The switch sends a login screen to the client webbrowser which must be authenticated before access is granted to the network. See **AAA** and **Tri-authentication**.

For a sample configuration script see **"Sample Web-Authentication Configuration" on page 66.9**.

Wildcard mask

A subnet mask in which bits set to 0 indicate an exact match and bits set to 1 indicate 'don't care'.