Chapter 17

IP Multicasting

Introduction

This chapter describes IP multicasting and support for multicasting on the router.

Most IP packets are sent to a single host – unicast transmission – or to all hosts on a network or subnetwork – broadcast transmission. Multicasting is an alternative where packets are sent to a group of hosts simultaneously on a network or sub-network. Multicasting is also known as *group transmission*.

A multicast environment consists of senders (IP hosts), routers and switches (intermediate forwarding devices) and receivers (IP hosts). A multicast group has a class D IP address (the first number in the IP address – the top four bits – are 1110). Any IP host can send packets to a multicast group, in the same way that they send unicast packets to a particular IP host, by specifying its IP address. A host need not belong to a multicast group in order to send to it. Packets sent to a group address are only received by members of the group.

The router uses the Internet Group Management Protocol (IGMP) to track multicast group membership, and one or more of the following protocols to route multicast traffic:

- Distance Vector Multicast Routing Protocol (DVMRP)
- Protocol Independent Multicast Sparse Mode (PIM-SM)
- Protocol Independent Multicast Dense Mode (PIM-DM)

For simple networks, the switch can be configured to use IGMP proxy instead of DVMRP or PIM.

The multicast routing protocols described in this chapter are dynamic and respond to changes in multicast group membership. Interfaces on the router can instead be configured statically to send and/or receive multicast packets. Static multicasting is described in "Static Multicast Forwarding" on page 14-47 of Chapter 14, Internet Protocol (IP).

Some interface and port types mentioned in this chapter may not be supported on your router. The interface and port types that are available vary depending on your product's model, and whether an expansion unit (PIC, NSM) is installed. For more information, see the AR400 Series Router Hardware Reference.

References

Internet Draft "*Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*", Internet Engineering Task Force, PIM WG, 1 March 2002 (draft-ietf-pim-sm-v2-new-05).

Internet Draft "*Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)*", Internet Engineering Task Force, PIM WG, 15 February 2002 (draft-ietf-pim-dm-new-v2-01).

RFC 2236 "*Internet Group Management Protocol, version 2*", November 1997.

Internet Draft "*Distance Vector Multicast Routing Protocol Version 3*", Internet Engineering Task Force, August 2000 (draft-ietf-idmr-dvmrp-v3-10).

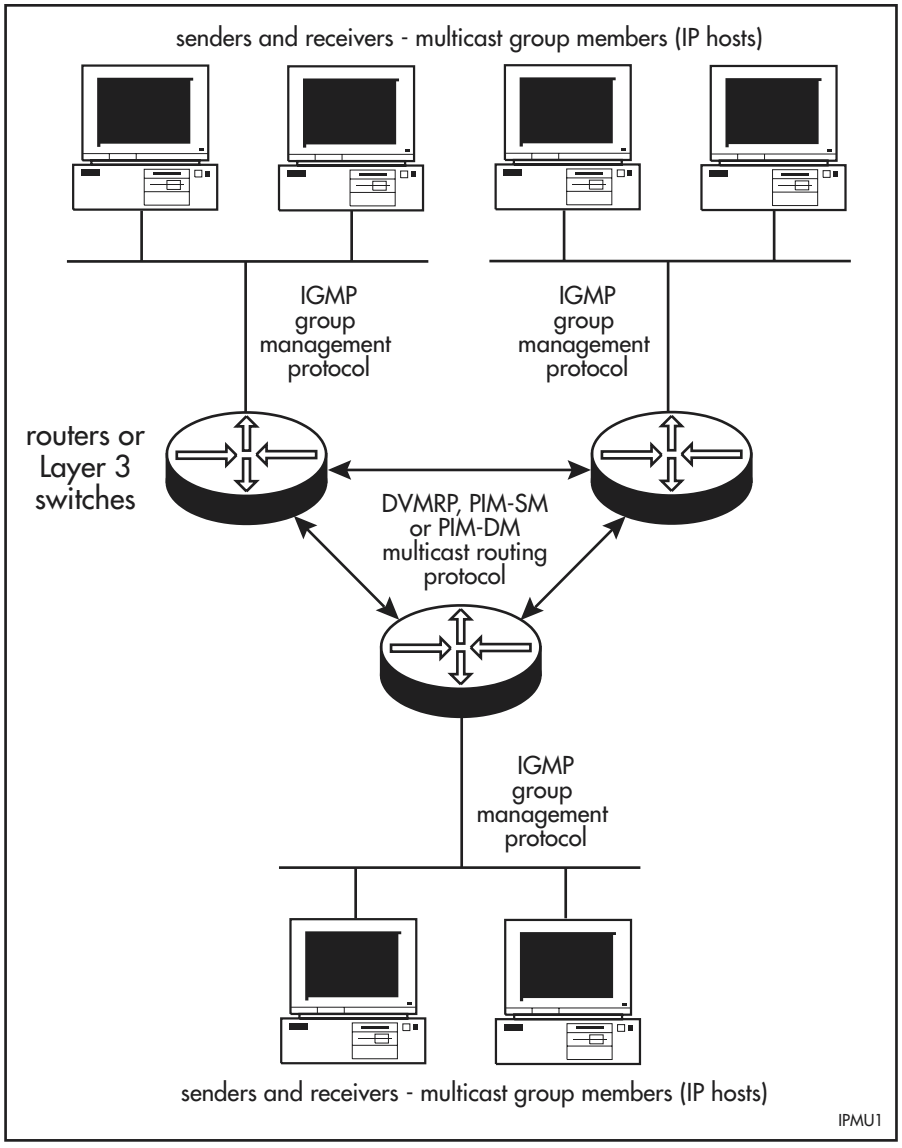
RFC 2715 "*Interoperability Rules for Multicast Routing Protocols*", October 1999.

IP Multicast Routing

For multicasting to succeed, the router needs to know which of its interfaces are directly connected to members of each multicast group. To establish this, the router uses IGMP for multicast group management ("Internet Group Management Protocol (IGMP)" on page 17-20). The router also needs to know which other routers to route multicast traffic to. The router maintains a routing table for multicast traffic with DVMRP ("Distance Vector Multicast Routing Protocol (DVMRP)" on page 17-5), PIM–Sparse Mode, or PIM-Dense Mode ("Protocol Independent Multicast (PIM)" on page 17-7). IGMP and one of the multicast routing protocols must be configured before the router can forward multicast packets.

The relationships between IP hosts, routers, and multicasting protocols are shown in the following figure.

Figure 17-1: Multicast environment.



When the router receives a packet addressed to a multicast group, it forwards it to the interfaces that have group members connected to them, according to IGMP, and out other interfaces specified by the multicast routing protocol. Membership in a multicast group is dynamic; hosts can join and leave at any time. Multicast groups can be long or short lived, and can have relatively stable or constantly changing membership. There is no limit on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time.

When the router finds out from IGMP that a new host has joined a multicast group on one of its interfaces, the router needs to receive the multicast traffic for this group, so that it can forward it to the host. The router uses the multicast routing protocol (DVMRP, PIM-SM or PIM-DM) to notify routers closer to the sender (upstream) to forward it traffic for the group. Routers running a multicast routing protocol, such as Protocol-Independent Multicast (PIM), maintain forwarding tables to forward multicast packets. DVMRP, PIM Sparse Mode and PIM Dense Mode share a multicast forwarding table.

Interoperability between Multicast Routing Protocols

The router can be configured as a Multicast Border Router (MBR), as specified in RFC 2715 "*Interoperability Rules for Multicast Routing Protocols*", October 1999. A Multicast Border Router forms the border between two or more multicasting domains that are running different multicast routing protocols (DVMRP, PIM-SM or PIM-DM). The MBR forwards multicast packets across the different domains so that receivers in one domain can receive packets from sources in another domain. Therefore different interfaces on the router can be configured as DVMRP, PIM-SM or PIM-DM interfaces.

The router treats sources that are reached via another multicasting domain as if they were directly connected sources.

Distance Vector Multicast Routing Protocol (DVMRP)

DVMRP is an Internet routing protocol that provides an efficient mechanism for connectionless datagram delivery to a group of hosts across an internetwork. It is a distributed protocol that dynamically generates IP Multicast delivery trees using a technique called Reverse Path Multicasting (RPM). The router supports DVMRP version 3 as specified in Internet Draft *"Distance Vector Multicast Routing Protocol Version 3', September 1999* (draft-ietf-idmr-dvmrp-v3-09).

DVMRP maintains its own multicast routing table, and uses this to maintain the shared multicast routing table. It maintains a list of its DVMRP neighbours, to which it sends and receives DVMRP messages.

DVMRP uses a distance vector distributed routing algorithm in order to build per-source-group multicast delivery trees. Datagrams follow multicast delivery trees from a source to all members of a multicast group, replicating the packet

only at necessary branches in the delivery tree. DVMRP calculates and updates the trees dynamically to track the membership of individual groups. When a datagram arrives on an interface, the reverse path to the source of the datagram is determined by examining a DVMRP routing table of known source networks. If the datagram arrives on an interface that would be used to transmit the datagram back to the source, then it is forwarded to the appropriate list of downstream interfaces. Otherwise, it is not on the optimal delivery tree and is discarded. In this way DVMRP filters out duplicate packets arising from loops in the network topology. DVMRP automatically prunes back the delivery tree for each source as group membership changes or routers determine that no group members are present. This keeps the delivery trees to the minimum branches necessary to reach all of the group members. New sections of the tree are added dynamically as new members join the multicast group by grafting the new sections onto the delivery trees.

The router sends *Prune* messages to the DVMRP router (neighbour) next closest to the multicast source when it thinks that it has no group members downstream dependent on it for this multicast group's traffic. If it later finds out that a new group member has joined downstream, it sends a *Graft* message to the upstream router to tell it that it needs to receive this multicast traffic again. These Join and Graft messages are propagated through the DVMRP network upstream towards the source of the multicast traffic as far as they are needed to keep the traffic flowing as far as it needs to, and no further.

The ports of a DVMRP router are physical interfaces to a directly attached subnetwork. All interfaces are configured with a metric that specifies the given port's part of the path cost, and a TTL threshold that limits the scope of a multicast transmission.

An IP interface has to be configured manually to run DVMRP by adding the IP interface to the DVMRP interface list.

Configuring DVMRP

DVMRP is disabled by default, and must be enabled on the router and on the interfaces it is to operate over to start multicast routing. DVMRP must be enabled on all the interfaces in the DVMRP domain over which the router sends or receives multicast data, including interfaces to directly connected IP hosts over which IGMP is enabled for group management. To enable or disable DVMRP on the router, use the commands:

```
enable dvmrp

disable dvmrp
```

To start DVMRP operating over an interface, add the interface to DVMRP using the command:

```
add dvmrp interface=interface [dlc=1..1024] [metric=1..32]
    [ttlthreshold=1..255]

delete dvmrp interface=interface [dlc=1..1024]
```

The IP configuration of an interface cannot be changed while DVMRP or PIM is attached to the interface. The DVMRP or PIM interface must first be deleted, and then re-added after the IP changes have been made.

Multicast packets are delivered along the shortest path from one host to another. The distance is the sum of metrics along this path. The router uses the path cost for routes on each interface to determine which interfaces to send

multicast traffic over. When the router receives a multicast packet over an interface, it adds the metric for the interface to the path cost. A higher metric for an interface causes less traffic to be transmitted across the interface. The default for the metric is 1, and this can be modified using the command:

```
set dvmrp interface=interface [dlc=1..1024] metric=1..32
```

When a router receives a multicast datagram, it forwards the datagram if its Time to Live (TTL) is less than the **ttlthreshold** parameter. The TTL is decremented at every hop. The default of the Time To Live Threshold is 255, and this can be modified with the command:

```
set dvmrp interface=interface [dlc=1..1024]
   ttlthreshold=1..255
```

To display information about the DVMRP state, interfaces, neighbours, routes or counters, use the commands:

```
SHow DVMrp

SHow DVMrp INTerface

SHow DVMrp NEIghbour

SHow DVMrp ROUte

SHow DVMrp COUnters
```

To reset all DVMRP processes, timers and route information for an interface, that is, to restart all DVMRP processes for an interface as if it has just been added to the DVMRP interface list, use the command:

```
reset dvmrp interface=interface [dlc=1..1024]
```

For experienced users, detailed debugging information can be displayed about the output of neighbours, grafts, prunes, probes and reports. Note that DVMRP debugging can display large amounts of data. To enable or disable DVMRP debugging, use the commands:

```
enable dvmrp debug={all|graft|neighbour|probe|prune|
   report}[,...] interface=interface [dlc=1..1024]

disable dvmrp debug={all|graft|neighbour|probe|prune|
   report}[,...] interface=interface [dlc=1..1024]
```

Protocol Independent Multicast (PIM)

The two Protocol Independent Multicast routing protocols rely on the presence of an existing unicast routing protocol to adapt to topology changes, but are independent of the mechanisms of the specific unicast routing protocol. PIM Dense Mode is most suitable for networks where bandwidth is plentiful, and where the members of a multicast group are densely distributed on the network. PIM Sparse Mode is more suitable when the members of the multicast groups are more sparsely distributed over the network, because it results in less duplication of data packets over the network.

The router can be configured as a Multicast Border Router, with different interfaces connecting to multicast domains that use different multicast routing protocols. Therefore, some PIM interfaces can be configured for PIM-SM and others for PIM-DM. Multicast packets are forwarded between the Sparse Mode and Dense Mode domains as required.

PIM Sparse Mode

PIM 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. It 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.

The router supports PIM Sparse Mode as specified in Internet Draft "*Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*", 1 March 2002 (draft-ietf-pim-sm-v2-new-05). For information about implementation of PIM-SM for IPv6, see "Protocol Independent Multicast Sparse Mode (PIM-SM)" on page 18-7 of Chapter 18, IPv6 Multicasting.

Routers with directly attached or downstream members are required to join a Sparse Mode distribution tree by transmitting explicit join messages. If a router 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 router, 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 routers 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.

PIM Sparse Mode Roles

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 router, and before the router can route multicast traffic, it must establish which of the PIM routers in the network are performing some key roles: *designated router* (DR), *Rendezvous Point* (RP), and *bootstrap router* (BSR).

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 designated router 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 routers 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 designated router for a multicast sender sends multicast packets towards the RP. Designated routers 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.

Note that previous release versions did not correctly support the PIM hash mask length option. As a result, the RP selection calculation differs between old and new release versions. If a network contains switches running a mixture of versions, this leads to incorrect forwarding behaviour. To avoid this issue, either ensure that all devices on the network correctly support the hash mask length option (recommended), or ensure that the following **both** hold:

- The hash mask length option on all BSR candidates is configured to 4 bits. This implies that all BSR candidates must be running a new release.
- All RP candidates use a common prefix of 224.0.0.0/240.0.0.0

This will have the side effect of collapsing all groups to use a single PIM RP.

Bootstrap Router

Each PIM-SM network must have at least one bootstrap router (BSR) candidate, unless all routers in the domain are configured statically with information about all RPs in the domain. Every router that is a BSR candidate periodically sends a Bootstrap Candidate Advertisement message to advertise that it is available as a bootstrap router candidate. The BSR candidates in the network elect the router with the highest preference value to be the bootstrap router. The elected bootstrap router 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 Rendezvous Point candidates. Each router uses this information and a standardised hash mechanism to determine the RP for each group.

In summary:

- Each *multicast group* must have at least one Rendezvous Point candidate
- Each *PIM-SM domain* must have at least one Bootstrap Router 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 Designated Router candidate.

PIM Hello Messages

When PIM is enabled on a router, it sends out a PIM *Hello* message on all its PIM enabled interfaces, and listens for Hello messages from its PIM neighbours. When a router receives a Hello message, it records the interface, IP address, priority for becoming a designated router, and the timeout for the neighbour's information. The router sends Hello messages regularly at the Hello Time interval.

PIM Sparse Mode Operation

Once these roles are established, multicast routing can operate in three phases. Phase one establishes and uses an RP Tree; phase two improves efficiency and performance using Register Stop; phase three further optimises routing by using Shortest Path Trees. While multicast routing always begins with phase one, the designated router for a receiver determines whether and when to move on to phases two and three, depending on the amount of traffic from the source.

Phase One – Rendezvous Point Tree

Phase one establishes and uses a shared tree rooted at the Rendezvous Point to forward all multicast data to group members.

When an IP host sends an IGMP Join message to the local PIM designated router, which is not the RP for the group, the designated router sends a PIM Join message towards the RP for the group ("upstream"). It determines which router is the RP for the group from the most recent bootstrap message. Every router it passes through records that there is a group member on the incoming interface. Eventually, the Join message reaches either the RP, or another router 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 an RP 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 designated router 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 cease, and their entries timeout on routers that are closer to the RP.

The sender's designated router encapsulates the multicast data in a unicast packet in a process called registering, and sends these register packets to the group's Rendezvous Point. When the RP receives the data, it decapsulates them, and forwards them onto the shared tree.

Phase Two – Register Stop

In phase two 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 the 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.

Phase Three – Shortest Path Tree

In phase three 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 Join message reaches the sender's DR, the sender's 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 shortest path tree (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 router that still needs to forward multicast packets from this sender to other receivers. Every time a router receives a Prune message, it waits a short time (the J/P Override Interval specified in Internet Draft draft-ietf-pim-sm-v2-new-05) before putting the Prune into effect, so that other routers 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 routers 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 routers 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 routers notice duplicate data packets on the LAN, they elect a single router to forward the data packets, by each sending PIM *Assert* messages. If one of the upstream routers is on an SPT and the other is on an RPT, the router on the SPT has the shortest path to the sender, and wins the Assert election. If both routers are on RPTs the router with the shortest path to the RP (the lowest sum of metrics to the RP) wins the Assert. If both routers are on an SPT, then the router with the shortest path to the sender (the lowest sum of metrics to the sender's DR) wins the Assert.

The router 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 routers on the LAN also receive the Assert messages, and send all their Join messages to the Assert winner. The result of an Assert election times out after the Assert Time specified in the Internet Draft draft-ietf-pim-sm-v2-new-05. 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 router 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 routers then switch back to the RP tree.

Configuring PIM Sparse Mode

PIM multicasting routing is disabled by default and must be enabled on the router before PIM configuration takes effect. However, we recommend that the PIM configuration be completely set up on the router before PIM is enabled. To enable or disable PIM, use the commands:

```
enable pim
```

```
disable pim
```

For PIM Sparse Mode multicast routing to operate on the router, each interface over which it is to send and receive multicast routing messages and multicast packets must be assigned to PIM-SM. Each subnetwork must also have at least one *Designated Router* candidate, each network must have at least one *Bootstrap Router* candidate, and each multicast group must have at least one *Rendezvous Point* candidate.

The IP configuration of an interface cannot be changed while DVMRP or PIM is attached to the interface. The DVMRP or PIM interface must first be deleted, and then re-added after the IP changes have been made.

PIM-SM Interfaces

By default PIM interfaces are set to use Sparse Mode when they are added. To add a PIM-SM interface, use the command:

```
add pim interface=interface [drpriority=0..4294967295]
    [electby={drpriority|ipaddress}] [mode=sparse]
    [other-options...]
```

Each PIM-SM interface has a priority for becoming the *designated router* (DR) for its subnetwork. The higher the number, the higher the priority. The default designated router priority is 1. If the multicast group must choose a DR from interfaces with the same priority, or no priority, the interface with the highest IP address number is chosen.

The **electby** parameter determines how the router elects the designated router for this interface. If **drpriority** is specified, the interface transmits its DR priority in its hello messages. If all routers in the subnetwork transmit their DR priorities, routers in the subnetwork can elect the DR by priority. If **ipaddress** is specified, the router does not transmit its DR priority, which forces the routers in the subnetwork to elect the DR by IP address. The default is **drpriority**.

To delete an interface, use the command:

```
delete pim interface=interface [dlc=1..1024]
```

To modify the mode, designated router priority, or method by which the designated router is elected for a PIM interface, use the command:

```
set pim interface=interface mode={dense|sparse}
    [drpriority=0..4294967295] [electby={drpriority|
    ipaddress}] [other-options...]
```

To restart all PIM processes on an interface, resetting the PIM timers, route information and counters for the interface, use the command:

```
reset pim interface=interface [dlc=1..1024]
```

To display information about PIM interfaces, use the command:

```
show pim interface
```

Bootstrap Router Candidates

Each network of PIM–SM routers must have a *bootstrap router (BSR)*. PIM-SM chooses as the Bootstrap Router the candidate with the highest preference value from all the Bootstrap Router candidates available. Each PIM–SM connected network must have at least one Bootstrap Router candidate. The candidate with the highest preference value becomes the Bootstrap Router. The default preference is 1. The Bootstrap Router sends a *bootstrap message* to the other PIM–SM routers, containing a list of the RP candidates for multicast groups at BSM interval seconds. To make the router a Bootstrap Router candidate, use the command:

```
add pim bsrcandidate [preference=0..255]
```

To change the router's Bootstrap Router candidate preference, use the command:

```
set pim bsrcandidate preference=0..255
```

To stop the router acting as a Bootstrap Router candidate, use the command:

```
delete pim bsrcandidate
```

To display information about the router's Bootstrap Router configuration, use the command:

```
show pim BSRCANDIDATE
```

Rendezvous Point

Each multicast group must have a *Rendezvous Point* (RP), which is either chosen dynamically from the list of rendezvous point candidates available, or statically configured on each router that processes traffic for that group. For dynamic RP selection, there must be at least one RP candidate in the PIM-SM connected network, but generally there should be several. PIM-SM chooses the RP candidate with lowest preference value to be the RP for the multicast group. The lower the number, the higher its priority. The default priority is 192. The dynamically-chosen RP advertises itself to the current Bootstrap Router at an interval specified by the **advinterval** parameter in the **set pim** command. The default **advinterval** is 60 seconds.

When an IP host joins a multicast group on a router, the router sends a *Join* message to the active rendezvous point. The rendezvous point then knows to send multicast packets for the group to this router. When the last IP host leaves a group, the router sends a *Prune* message to the RP, telling it that it no longer needs to receive multicast packets for the group.

To configure the router to be a dynamic RP candidate, use the command:

```
add pim rpcandidate group=group-address [mask=ipaddress]
   [priority=0..255]
```

To modify the router's RP candidate priority, use the command:

```
set pim rpcandidate group=ipadd [mask=ipadd] priority=0..255
```

*The router has the same values for **priority** for all multicast groups for which it is a rendezvous point candidate, so changing the priority for one group changes it for all groups.*

To stop the router from acting as an RP candidate, use the command:

```
delete pim rpcandidate group=group-address [mask=ipadd]
```

Static RP mappings can be configured instead of using the bootstrap mechanism. To configure a static Rendezvous Point on the router for a multicast group, specify the IP address of the Rendezvous Point, using the command:

```
add pim rpcandidate=rp-address group=group-address
    [mask=ipaddress]
```

where *rp-address* is the IP address of the router that is the Rendezvous Point for the multicast group(s) specified. An RP can be statically configured as the RP for multiple groups, but each group can only have one statically-configured RP. Each router in the PIM-SM domain must be configured with the same static RP to group mapping.

If the bootstrap mechanism is also running, a static RP mapping takes precedence.

To delete a static RP, use the command:

```
delete pim rpcandidate=rp-address group=group-address
    [mask=ipaddress]
```

To display information about multicast groups for which the router is a Rendezvous Point candidate, use the command:

```
show pim rpcandidate
```

To display the static group-to-RP mapping followed by the elected bootstrap router's current set of RP candidates and the groups they are configured for, use the command:

```
show pim rpset
```

General PIM-SM Information

The following commands display general PIM-SM information.

Command	Meaning
show pim config	Lists CLI commands that make up the router's PIM configuration
show pim counters	Displays the number of PIM messages that the router has received and sent, and the number of bad messages it has received
show pim neighbour	Gives information about the neighbouring routers that PIM is aware of
show pim route	Displays the internal PIM routing table

PIM-SM Timers

Timers for PIM-SM operations have defaults that suit most networks, and should not generally be modified. If they need to be modified, use the command:

```
set pim [advinterval={10..15000|default}]
    [bsminterval={10..15000|default}] [jpinterval={1..65535|
    default}] [keepalivetime={10..65535|default}]
    [probetime={1..65535|default}]
    [suppressiontime={1..65535|default}] [other-options...]
```

Changing these timers to inappropriate values can cause PIM to function in undesirable ways. System administrators should change these timer values based on a sound understanding of their interaction with other devices in the network.

To list the values of the global PIM timers, use the command:

```
show pim timer
```

PIM-SM Debugging

To display debugging information about PIM-SM, use the command:

```
enable pim debug={all|assert|bsr|c-rp-adv|hello|join|
    register}[,...]
```

The debugging that results from specifying each of these options is shown in Table 17-2 on page 17-55.

To see which debugging options are enabled, use the command:

```
show pim debug
```

Logging and SNMP Traps for PIM-SM

PIM-SM can now be configured to produce log messages in response to status changes and errors, and SNMP traps.

This enhancement does not apply to PIM-DM.

Status log messages

Events that will trigger a status-change log message are:

- PIM interface is disabled
- PIM interface is enabled
- PIM neighbour adjacency has timed out
- PIM neighbour generation ID has changed
- PIM neighbour has changed port
- PIM RP has changed
- PIM DR has changed
- PIM BSR has changed

Error log messages

Errors that will trigger a log message are:

- Invalid PIM packet
- Invalid destination address
- Fragmentation reassembly
- Packet too short
- Bad group address encoding
- Bad source address encoding
- Missing option
- Internal error
- Receive packet - a range of errors that mean the packet was received but cannot be forwarded.

SNMP traps

The following traps are sent:

- PimInterfaceUpTrap - generated when a PIM interfaces comes up and is active
- PimInterfaceDownTrap - generated when a PIM interfaces goes down and is in-active
- PimNeighbourLossTrap - generated when a known PIM neighbour has loss adjacency or has timed-out. This trap is part of the experimental PIM MIBs group
- PimNeighbourAddedTrap - generated when a PIM neighbour is added
- PimNeighbourDeletedTrap - generated when a PIM neighbour is deleted
- PimErrorTrap - generated when any one of the PIM error counters is incremented or when a log message of subtype LOG_STY_PIM_ERROR is generated (see list of errors above)

To specify the type of log messages and SNMP traps that the switch generates, use the command:

```
set pim log={none|status|error|all}
    [trap={none|status|error|all}]
```

To display the specified options, use the command:

```
show pim debug
```

PIM Dense Mode

Unlike PIM Sparse Mode, PIM Dense Mode (PIM-DM) does not use a designated router, bootstrap router, or Rendezvous Points.

PIM-DM is similar to DVMRP in that it employs the Reverse Path Multicasting (RPM) algorithm. However, there are differences between PIM-DM and DVMRP:

- PIM-DM relies on the presence of an existing unicast routing protocol to provide routing table information to build up information for the multicast forwarding database, but it is independent of the mechanisms of the specific unicast routing protocol. In contrast, DVMRP contains an integrated routing protocol that makes use of its own RIP-like exchanges to compute the required unicast routing information.
- Unlike DVMRP, PIM-DM simply forwards multicast traffic on all downstream interfaces until explicit prune (un-join) messages are received. PIM-DM is willing to accept the overhead of broadcast-and-prune in the interests of simplicity and flexibility, and of eliminating routing protocol dependencies.

PIM-DM assumes that when a source starts sending, all downstream systems want to receive multicast datagrams. Initially, multicast datagrams are flooded to all areas of the network. If some areas of the network do not have group members, dense-mode PIM prunes the forwarding branch by setting up prune state. The prune state has an associated timer, which on expiration turns into forward state, allowing data to go down the branch that was previously in prune state.

The prune state contains source and group address information. When a new member appears in a pruned area, a router can “graft” toward the source for the group, turning the pruned branch into a forwarding branch. The forwarding branches form a tree rooted at the source leading to all members of the group. This tree is called a source rooted tree.

The broadcast of datagrams followed by pruning of unwanted branches is often referred to as a broadcast-and-prune cycle, typical of dense mode protocols. The broadcast-and-prune mechanism in PIM Dense Mode uses a technique called *reverse path forwarding* (RPF), in which a multicast datagram is forwarded only when the receiving interface is the one used to forward unicast datagrams to the source of the datagram.

Configuring PIM Dense Mode

PIM multicasting routing is disabled by default and must be enabled on the router before any PIM configuration takes effect. However, we recommend that the PIM configuration be completely set up on the router before PIM is enabled. To enable or disable PIM, use the commands:

```
enable pim
```

```
disable pim
```

For PIM Dense Mode multicast routing to operate on the router, each interface over which it is to send and receive multicast routing messages and multicast packets must be assigned to PIM-DM.

By default PIM interfaces are set to use Sparse Mode when they are added. To add a PIM-DM interface, use the command:

```
add pim interface=interface mode=dense [other-options...]
```

To delete an interface, use the command:

```
delete pim interface=interface [dlc=1..1024]
```

The IP configuration of an interface cannot be changed while DVMRP or PIM is attached to the interface. The DVMRP or PIM interface must first be deleted, and then re-added after the IP changes have been made.

To modify a PIM interface, use the command:

```
set pim interface=interface [mode={dense|sparse}]
   [other-options...]
```

State Refresh messages can be used in a PIM-DM domain to reduce unnecessary multicast traffic. Instead of a source repeatedly flooding downstream routers with multicast packets and repeatedly receiving prune messages, a State Refresh message maintains an existing prune. By default the router cannot initiate or process State Refresh messages. To enable this functionality on an interface, use one of the commands:

```
add pim interface=interface mode=dense srcapable=yes
   [other-options...]
```

```
set pim interface=interface srcapable=yes [other-options...]
```

To restart all PIM processes on an interface, resetting the PIM timers, route information and counters for the interface, use the command:

```
reset pim interface=interface [dlc=1..1024]
```

To display information about PIM interfaces, use the command:

```
show pim interface
```

General PIM-DM Information

The following commands display general PIM-DM information.

Command	Meaning
show pim config	Lists CLI commands that make up the router's PIM configuration
show pim counters	Displays the number of PIM messages that the router has received and sent, and the number of bad messages it has received
show pim neighbour	Gives information about the neighbouring routers that PIM is aware of
show pim route	Displays the internal PIM routing table
show pim staterefresh	Displays the internal State Refresh table

PIM-DM Timers

Timers for PIM-DM operations have defaults that suit most networks and should not generally be modified. If they need to be modified, use the command:

```
set pim [jpinterval={1..65535|default}]
    [keepalivetime={10..65535|default}]
    [pruneholdtime={1..65535|default}]
    [sourcealivetime={10..65535|default}]
    [srinterval={10..255|default}] [other-options...]
```

Changing these timers to inappropriate values can cause PIM to function in undesirable ways. System administrators should change these timer values based on a sound understanding of their interaction with other devices in the network.

To list the values of the global PIM timers, use the command:

```
show pim timer
```

PIM-DM Debugging

To display debugging information about PIM-DM, use the command:

```
enable pim debug={all|assert|graft|hello|join|
    staterefresh}[,...]
```

The debugging that results from specifying each of these options is shown in Table 17-2 on page 17-55.

To see which debugging options are enabled, use the command:

```
show pim debug
```

Internet Group Management Protocol (IGMP)

IGMP is a protocol used between hosts and multicast routers and switches on a single physical network to establish hosts' membership in particular multicast groups. Multicast routers use this information, in conjunction with a multicast routing protocol, to support IP multicast forwarding across the Internet.

The router supports Internet Group Management Protocol version 2 (IGMPv2), defined in RFC 2236 "Internet group Management Protocol, version 2", November 1997. It can also detect and interoperate with hosts and other designated routers (sometimes called querier routers) running IGMP version 1.

When IGMP is enabled on the router, and on particular interfaces, it sends out IGMP queries on all IGMP interfaces. If it receives an IGMP message from a router with a lower IP address on an interface, it knows that another router is acting as the IGMP designated router for that subnetwork. If it receives no IGMP messages with a lower IP address, it takes the role of designated router for that subnetwork. If it is the designated router, it continues to send out general IGMP *Host Membership Queries* regularly on this interface.

When an IP host hears a general IGMP Host Membership Query from the router, it sends an IGMP *Host Membership Report* back to the router. All the IGMP routers on the subnetwork put an entry into their *local group database*, so that the routers know which interfaces to send packets for this multicast group out. These entries are updated regularly, as long as the interface has a member of the multicast group connected to it. As hosts join and leave multicast groups dynamically, the router keeps a list of group memberships for each of its primary interfaces. In the case of multihomed interfaces, the primary interface is the first interface to be configured ("Multihoming" on page 14-11 of Chapter 14, Internet Protocol (IP)).

When an IP host stops belonging to a multicast group, it sends an IGMP *Leave* message to the router. The router then sends a group-specific IGMP membership query, and any other IP hosts belonging to the same multicast group reply with a Host Membership Report. IGMP then knows whether there are still any members of this multicast group connected to the interface.

Static IGMP

It is possible to have a network segment that either has no multicast group members, or has a host that is unable to report its group membership with IGMP. In such cases, no multicast traffic is sent to the network segment. This means that you need to pull down multicast traffic to the segment.

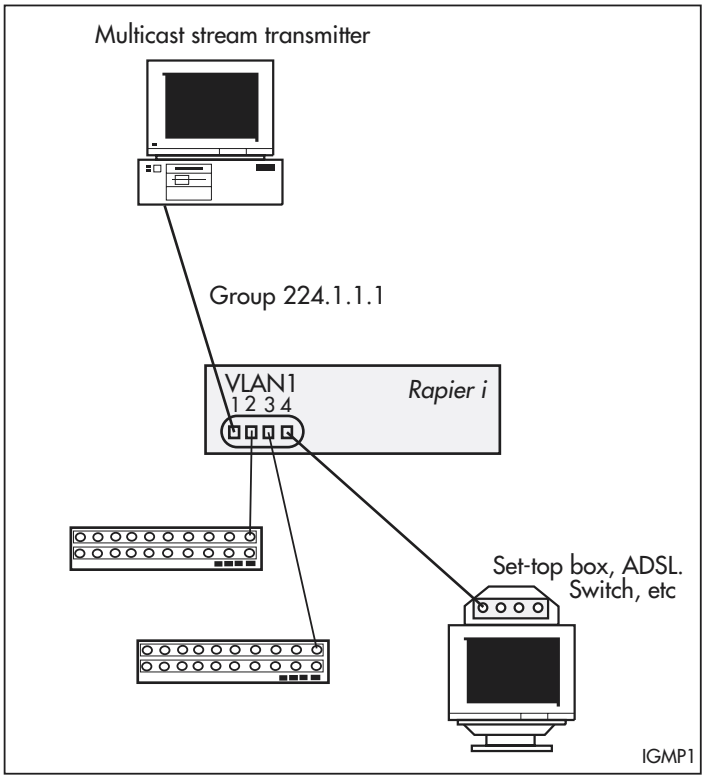
The user to instructs the switch to forward multicast data over a specified interface and port, as shown in Figure 17-2 on page 17-21. This capability is essential for sending multicast traffic to hosts that cannot report their group membership with IGMP. It plays an important role in video over ADSL applications.

Figure 17-2 on page 17-21 illustrates a switch forwarding the multicast stream to a set-top box after a user specifies that group 224.1.1.1 multicast data should be forwarded out of port 4 of VLAN1.

Unlike conventional IGMP membership, this user-specified *static membership* never times out.

You can also filter some IGMP debug messages by source IP address and group destination address.

Figure 17-2: Forwarding multicast data over a specified interface and port.



Configuring IGMP

IGMP is disabled by default on the router, and on all interfaces. To enable or disable IGMP on the router, use the commands:

```
enable ip igmp
disable ip igmp
```

IGMP must be enabled on an interface before it can send or receive IGMP messages on the interface. If DVMRP is used for multicast routing, IGMP must also be enabled on interfaces that DVMRP uses. To enable or disable IGMP on an interface, use the commands:

```
enable ip igmp interface=interface
disable ip igmp interface=interface
```

IGMP keeps the local group database up to date with current multicast group members by updating it when it hears IGMP Host Membership Reports on an interface. If the router is the IGMP designated router for the subnetwork, it sends out IGMP Host Membership Queries at a Query Interval. If it does not receive a Host Membership Report for a multicast group on an interface within the Timeout period, it deletes the multicast group from its local group database. The default of the Query Interval (125 seconds) and of the Timeout (2*(Query Interval + 10) seconds) suit most networks. These defaults should be

changed with caution, and with a sound understanding of how they affect interaction with other devices. To change the intervals, use the command:

```
set ip igmp [lmqi=1..255] [lmqc=1..5]
   [queryinterval=1..65535] [queryresoibseinterval=1..255]
   [robustness=1..5] [timeout=1..65535]
```

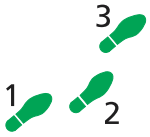
The command:

```
show ip igmp
```

displays information about IGMP and multicast group membership.

Configuring Static IGMP

The following configuration example illustrates the steps required to create a static IGMP association. It assumes that *vlan1* has already been configured as an IP interface on the switch.



Configure a static IGMP association

1. **Enable IGMP on the switch.**

   ```
   enable ip igmp
   ```

2. **Enable IGMP on vlan1.**

   This must be done before the static IGMP association is created.

   ```
   enable ip igmp interface=vlan1
   ```

3. **Create the static IGMP association.**

   The multicast data for the group specified by the **destination** parameter is forwarded over ports specified by the **port** parameter. If the **port** parameter is not entered, the association defaults to all ports belonging to the interface.

   ```
   create ip igmp destination=224.1.2.3 interface=vlan1
      port=1-4
   ```

4. **Check the configuration.**

   Check that the static IGMP association has been created and IGMP is enabled.

   ```
   show ip igmp destination=224.1.2.3 interface=vlan1
   ```

IGMP Snooping

On the AR450S, IGMP snooping lets routers intelligently forward multicast traffic instead of flooding all ports in the VLAN. Because IGMP is an IP-based protocol, multicast group membership for VLAN-aware devices is on a per-VLAN basis. If at least one port in the VLAN is a member of a multicast group and IGMP snooping is not used, multicast packets flood all ports in the VLAN.

With IGMP snooping, the router listens to IGMP membership reports, queries, and leave messages to identify ports that are members of multicast groups. Multicast traffic is forwarded to ports that are members of a specific multicast group.

IGMP snooping happens automatically at Layer 2 on VLAN interfaces. By default, the router forwards traffic from ports with multicast listeners, and does not act as a simple hub and flood multicast traffic from 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.

To set operation for IGMP snooping, use the command:

```
set igmpsnooping routermode=[all|default|ip|multicastrouter|
   none]
```

If **all** is specified, all reserved multicast addresses (i.e. 224.0.0.1 to 224.0.0.255) are treated as router multicast addresses. If **default** is specified, a specific list is used as router multicast addresses (Table 17-3 on page 17-60). If **ip** is specified, users can specify addresses to be treated as router multicast addresses. If **multicastrouter** is specified, 224.0.0.4 is treated as a multicast address for DVMRP routers, and 224.0.0.13 for all PIM routers.

To add and delete reserved IP multicast addresses to and from the list of router multicast addresses that are specified by the **set igmpsnooping routermode** command when the **ip** parameter is selected, use the commands:

```
add igmpsnooping routeraddress
```

```
delete igmpsnooping routeraddress
```

The IP addresses specified must be from 224.0.0.1 to 224.0.0.255.

To display information about the current list of configured IP multicast router addresses configured on the router, use the command:

```
.show igmpsnooping routeraddress
```

IGMP snooping is enabled by default. To disable it, use the command:

```
disable igmpsnooping
```

Disabling IGMP snooping may be useful when filters are used extensively because IGMP snooping uses a Layer 3 filter. When IGMP snooping is disabled, this filter becomes available. Note that multicast packets flood the VLAN when IGMP snooping is disabled.

To enable IGMP snooping, use the command:

```
enable igmpsnooping
```

IGMP snooping can be enabled only when a free filter entry is available.

To display information about IGMP snooping, use the command:

```
show igmpsnooping [vlan=vlan] [counter]
```

IGMP Snooping All-groups

IGMP snooping all-groups allows you to prevent a port or ports from acting as an all-groups entry.

Sometimes the device cannot differentiate between certain multicast addresses and permanent host groups at Layer 2. For example, this happens with the addresses 239.0.0.2 and 224.0.0.2 where 224.0.0.2 is the all-routers multicast group. If the device receives an IGMP report for the 239.0.0.2 address, which has a MAC address of 01-00-5e-00-00-02, the device will create an all-groups entry in the MARL. All further multicast groups will be added to this port, so multicast traffic will be forwarded out the port.

By preventing a port or ports from receiving an all-groups entry, you can limit the number of router ports on the device, and therefore the volume of multicast traffic sent over the device's ports. Once disabled with the **disable ip igmp allgroup** command, the port will no longer create MARL entries when the device receives an IGMP report, query, or multicast data over any other port. For example, if port 9 has been disabled as an all-groups port, an all-groups entry will be created for port 9. This will happen when the port receives packets that will create an IGMP router port, such as reserved multicast groups and IGMP queries. However, a subsequent IGMP report received over port 7 will have an entry made for port 7 only. The IGMP group received on port 7 will not be added to port 9.

The all-groups disabled ports can be viewed in the output of the **show ip igmp** and **show igmpsnooping** commands.

IGMP Proxy

If the network topology is a simple tree, IGMP proxying can be used instead of a multicast routing protocol. The router at the root of the tree runs a multicast routing protocol. Other routers in the tree receive IGMP messages from their downstream interfaces and proxy them up the tree via their upstream interface.

A downstream interface is an interface that is not in the direction of the root of the tree. IGMP proxying works on routers with any number of downstream interfaces.

An upstream interface is in the direction of the root of the tree. IGMP proxying works only on routers with one upstream interface.

IGMP messages received from any downstream interfaces are proxied to the appropriate upstream interface. The router maintains a list of interfaces in each multicast group. A group is based on the multicast destination address. When a multicast group is created as a result of an interface receiving an IGMP join message, the router sends a IGMP join message via its upstream interface.

The IGMP proxy does not respond to IGMP join or leave messages received via its upstream interface. It responds to IGMP query messages received via its upstream interface.

The IGMP proxy sends an IGMP leave message via its upstream interface when the last interface has left the group.

Multicast packet forwarding is enabled if a multicast routing protocol is not already enabled, and when the router has an interface configured with IGMP proxy in the upstream direction and at least one interface configured with IGMP proxy in the downstream direction.

To add an IP interface and specify IGMP proxying (**add ip interface** command on page 14-77 of Chapter 14, Internet Protocol (IP)), use the command:

```
add ip interface=interface ipaddress={ipadd|dhcp}
   [igmpproxy={off|upstream|downstream}] [other-options...]
```

To set an IP interface with the IGMP proxy option specified, use the command (page 14-145 in Chapter 14, Internet Protocol (IP)):

```
set ip interface=interface
   igmpproxy={off|upstream|downstream}]
```

IGMP proxying is OFF by default. IGMP must be enabled on an interface for IGMP proxy to be configured to something other than OFF.

To enable IGMP on a specified interface, use the command:

```
enable ip igmp interface=interface
```

IGMP must also be globally enabled. To enable IGMP, use the command:

```
enable ip igmp
```

To display information about IGMP and IGMP proxy, use the command:

```
show ip igmp
```

Configuration Examples

This section includes examples of multicasting configuration using IGMP to manage group memberships for hosts directly connected to a router, and DVMRP, PIM-SM, or PIM-DM for multicast routing between routers.

Multicasting using DVMRP

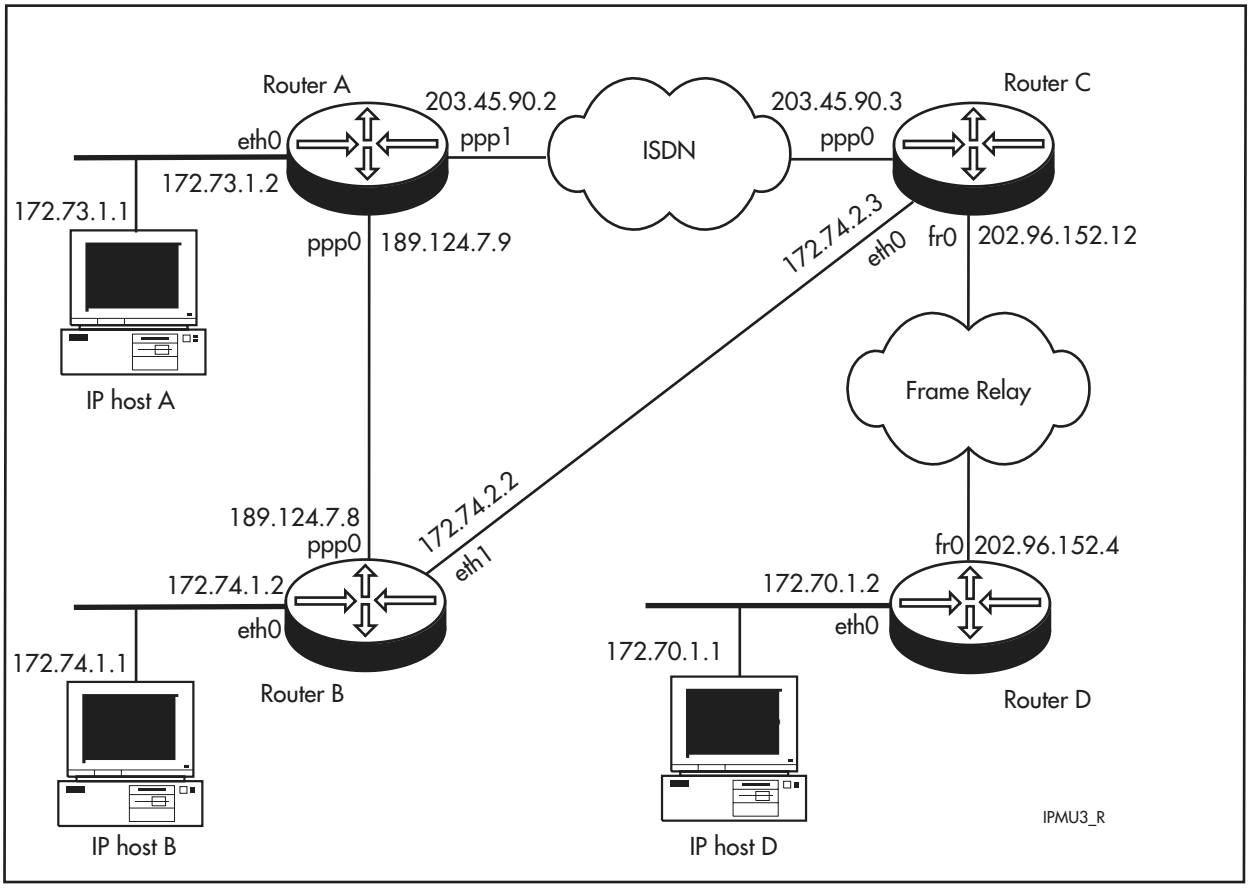
This example allows IP hosts to send data to and receive data from the multicast groups. Multicast group management uses IGMP, and multicast routing between the routers uses DVMRP. The example assumes that each router starts from the default configuration.

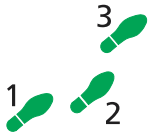
Some interface and port types mentioned in this example may not be supported on your router. The interface and port types that are available vary depending on your product's model, and whether an expansion unit (PIC, NSM) is installed. For more information, see the AR400 Series Router Hardware Reference.

Multicast packets are delivered along the shortest path from one host to another. The distance is the sum of metrics along this path. So in this example, the shortest path from IP host A to IP host B is Router A → Router C → Router B. From IP host A to IP host D the shortest path is Router A → Router C → Router D. If IP host B joins the multicast group to which IP host A is a sender, multicast data packets are not delivered to Router D or IP host D, unless IP host D also joins the same multicast group. Changing the metric on interfaces may change the path by which multicast packets are delivered.

Interfaces with DVMRP enabled must also have IGMP enabled.

Figure 17-3: Multicast configuration example using IGMP and DVMRP.





Configure multicast routing using DVMRP on Router A

1. **Set the system name.**

   Set a unique system name for the router.

   ```
   set sys name=A-dvmrp
   ```

2. **Configure ISDN.**

   Set up an ISDN call to Router C for DVMRP multicast traffic.

   ```
   add isdn call=dvmrp number=1234567 precedence=out
      outsub=local searchsub=local
   ```

3. **Configure PPP.**

   Create PPP interfaces over a synchronous port and the ISDN call.

   ```
   create ppp=0 over=syn0
   ```

   ```
   create ppp=1 over=isdn-dvmrp idle=on
   ```

4. **Configure IP.**

   Enable the IP module, and assign IP addresses to the interfaces.

   ```
   enable ip
   ```

   ```
   add ip interface=PPP0 ipaddress=189.124.7.9
      mask=255.255.0.0
   ```

   ```
   add ip interface=PPP1 ipaddress=203.45.90.2
      mask=255.255.255.0
   ```

   ```
   add ip interface=ETH0 ipaddress=172.73.1.2
      mask=255.255.255.0
   ```

5. **Configure IGMP.**

   Enable IGMP on the router for multicast group management.

   ```
   enable ip igmp
   ```

   Enable IGMP on the interfaces that have potential multicast receivers (IP hosts) connected to them, and the interfaces using DVMRP.

   ```
   enable ip igmp interface=eth0
   ```

   ```
   enable ip igmp interface=ppp0
   ```

   ```
   enable ip igmp interface=ppp1
   ```

6. **Configure DVMRP.**

   Enable DVMRP for multicast routing.

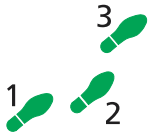
   ```
   enable dvmrp
   ```

   Enable DVMRP on the interfaces that use DVMRP for multicast routing. Setting the metrics on each of the interfaces influences the path cost and therefore the traffic sent over the interface. (The higher the metric, the higher the path cost, and the lower the traffic over the interface.)

   ```
   add dvmrp interface=eth0 metric=1
   ```

   ```
   add dvmrp interface=ppp0 metric=6
   ```

   ```
   add dvmrp interface=ppp1 metric=3
   ```



Configure multicast routing using DVMRP on Router B

1. **Set the system name.**

   Set a unique system name for the router.

   ```
   set sys name=B-dvmrp
   ```

2. **Configure PPP.**

   Create a PPP interface over a synchronous port.

   ```
   create ppp=0 over=syn0
   ```

3. **Configure IP.**

   Enable IP on the router, and assign IP addresses to the interfaces used by DVMRP for multicast routing.

   ```
   enable ip

   add ip interface=ppp0 ipaddress=189.124.7.8
      mask=255.255.0.0

   add ip interface=eth0 ipaddress=172.74.1.2
      mask=255.255.255.0

   add ip interface=eth1 ipaddress=172.74.2.2
      mask=255.255.255.0
   ```

4. **Configure IGMP.**

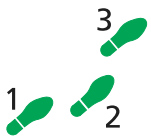
   Enable IGMP on the router, and on the interfaces that have IP host connected to them, so that the router can maintain its group membership data.

   ```
   enable ip igmp

   enable ip igmp interface=ppp0

   enable ip igmp interface=eth0

   enable ip igmp interface=eth0
   ```

5. **Configure DVMRP**

   Enable DVMRP on the router and on each interface over which it is used for multicast routing.

   ```
   enable dvmrp

   add dvmrp interface=eth0 metric=1

   add dvmrp interface=eth1 metric=1

   add dvmrp interface=ppp0 metric=6
   ```



Configure multicast routing using DVMRP on Router C

1. **Set the system name.**

   Set a unique system name for the router.

   ```
   set sys name=C-dvmrp
   ```

2. **Configure Frame Relay.**

   Configure a Frame Relay interface over a synchronous port to Router D, and add a data link circuit to the Frame Relay interface.

   ```
   create framerelay=0 over=syn0 lmischeme=none

   add framerelay=0 dlc=20
   ```

3. **Configure ISDN.**

   Set up an ISDN call to Router A for DVMRP multicast traffic. This call must have the same name as the ISDN call from Router A, and the opposite precedence.

```
add isdn call=dvmrp outsub=LOCAL searchsub=local
   precedence=in num=7654321
```

4. **Configure PPP.**

   Configure a PPP interface over the ISDN interface.

```
create ppp=0 over=isdn-dvmrp idle=on
```

5. **Configure IP.**

   Enable the IP module, and assign IP addresses to the interfaces.

```
enable ip

add ip interface=fr0 ipaddress=202.96.152.12
   mask=255.255.255.0

add ip interface=ppp0 ipaddress=203.45.90.3
   mask=255.255.255.0

add ip interface=eth0 ipaddress=172.74.2.3
   mask=255.255.255.0
```

6. **Configure IGMP.**

   Enable IGMP on the router and on the interfaces over which group membership is to be managed.

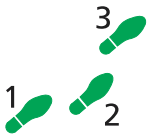
```
enable ip igmp

enable ip igmp interface=eth0

enable ip igmp interface=ppp0

enable ip igmp interface=fr0
```

   IGMP snooping is enabled by default, so does not need to be configured.

7. **Configure DVMRP.**

   Enable DVMRP on the router, and assign the interfaces over which DVMRP performs multicast routing.

```
enable dvmrp

add dvmrp interface=eth0vlan4 metric=1

add dvmrp interface=ppp0 metric=3

add dvmrp interface=fr0 dlc=20 metric=6
```



Configure multicast routing using DVMRP on Router D

1. **Set the system name.**

   Set a unique system name for the router.

```
set sys name=D-dvmrp
```

2. **Configure Frame Relay.**

   Create a Frame Relay interface over a synchronous port to Router C, and add a data link circuit to the Frame Relay interface.

```
create fr=0 over=syn0 lmi=none

add fr=0 dlc=20
```

3. **Configure IP.**

   Enable IP, and assign IP addresses to the interfaces.

```
enable ip
add ip interface=eth0 IP=172.70.1.2 mask=255.255.255.0
add ip interface=fr0 IP=202.96.152.4 mask=255.255.255.0
```

4. **Configure IGMP.**

   Enable IGMP on the router, and on the interfaces over which group membership is managed.

```
enable ip igmp
enable ip igmp interface=eth0
enable ip igmp interface=fr0
```

5. **Configure DVMRP.**

   Enable DVMRP on the router and on the interfaces over which DVMRP performs multicast routing.

```
enable dvmrp
add dvmrp interface=eth0 metric=1
add dvmrp interface=fr0 dlc=20 metric=6
```

Confirm multicasting

When the three routers have been configured, the DVMRP route exchange takes a few seconds. Then the IP hosts connected to these interfaces can send and receive multicasts.

1. **Test multicasting.**

   Test whether IP multicasting is successful by sending IP multicast data between hosts connected to each of the routers. Check that IGMP report and leave messages are correctly processed by having hosts leave and join groups, and check that IP multicast data forwarding stops and starts correctly.

2. **Check the multicast state.**

   To check each router, use the commands:

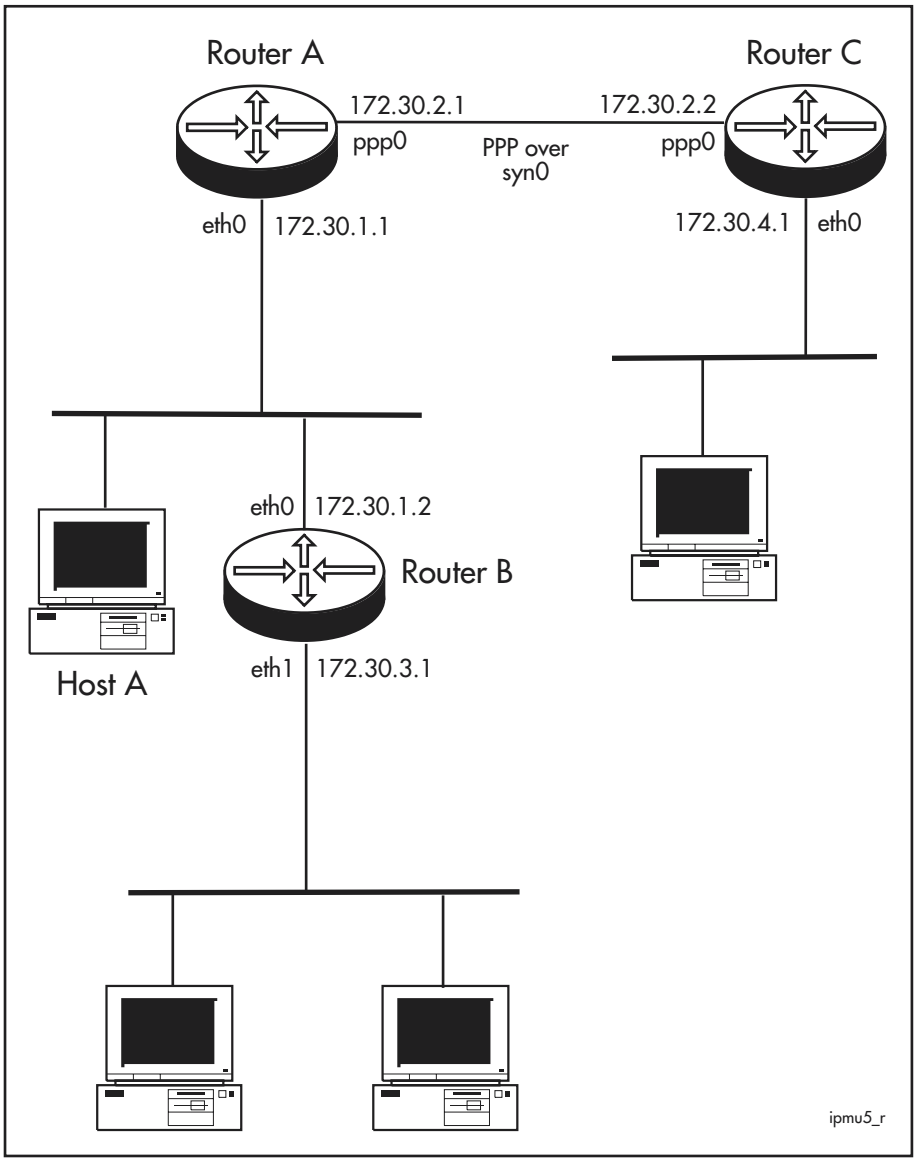
```
show dvmrp
show ip igmp
show ip route multicast
```

Protocol Independent Multicast (PIM)

These examples use PIM-SM or PIM-DM for multicast routing between three routers. The network topology is the same for each example (see Figure 17-4 on page 17-31). Multicast group management uses IGMP. The examples assume that each router starts from the default configuration.

Some interface and port types mentioned in these examples may not be supported on your router. The interface and port types that are available vary depending on your product's model, and whether an expansion unit (PIC, NSM) is installed. For more information, see the AR400 Series Router Hardware Reference.

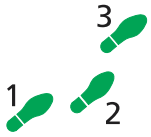
Figure 17-4: Multicast configuration topology using PIM Sparse or Dense Mode.



PIM Sparse Mode

This example allows IP hosts to send data to and receive data from the multicast groups 225.1.0.0 to 225.1.0.255. The configuration of Routers A, B and C are very similar, but Router A is the only router in this example that is configured as a PIM Bootstrap Router Candidate and a PIM Rendezvous Point Candidate.

Some interface and port types mentioned in this example may not be supported on your router. The interface and port types that are available vary depending on your product's model, and whether an expansion unit (PIC, NSM) is installed. For more information, see the AR400 Series Router Hardware Reference.



Configure multicast routing using PIM Sparse Mode on Router A

1. **Set the system name for the router.**

```
set sys name=A-pim-rp
```

2. **Create a PPP interface.**

Create a PPP interface 0 over synchronous port 0.

```
create ppp=0 over=syn0
```

3. **Configure IP.**

Enable IP and assign IP addresses for the PPP and Ethernet interfaces on the router.

```
enable ip

add ip interface=eth0 ipaddress=172.30.1.1
   mask=255.255.255.0

add ip interface=ppp0 ipaddress=172.30.2.1
   mask=255.255.255.0
```

4. **Configure a unicast routing protocol.**

Enable RIP over all interfaces.

```
add ip rip int=eth0

add ip rip int=ppp0
```

5. **Configure IGMP.**

Enable IGMP on the router for group management.

```
enable ip igmp
```

Enable IGMP on each interface, so that IGMP can find which multicast groups have hosts connected to the interfaces.

```
enable ip igmp interface=eth0

enable ip igmp interface=ppp0
```

6. **Configure PIM.**

Define PIM interfaces for the Ethernet and PPP interfaces.

```
add pim interface=eth0

add pim interface=ppp0
```

The network must have a PIM bootstrap router, so at least one router in the network must be configured as a Bootstrap Router Candidate. Set this router to be the Bootstrap Router Candidate.

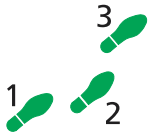
```
add pim bsrcandidate
```

At least one router in each multicast group must be a PIM Rendezvous Point (RP) for the multicast group, so at least one router in each group must be configured as a Rendezvous Point Candidate. Set this router to be an RP Candidate.

```
add pim RPCANDIDATE GROUP=225.1.0.0 mask=255.255.255.0
```

Enable PIM multicast routing.

```
enable pim
```



Configure multicast routing using PIM Sparse Mode on Router B

1. **Set the system name.**

   Set a unique system name on the router.

   ```
   set sys name=B-pim
   ```

2. **Configure IP.**

   Enable IP, and assign IP addresses to the Ethernet interfaces.

   ```
   enable ip
   add ip interface=eth0 ipaddress=172.30.1.2 mask=
      255.255.255.0
   add ip interface=eth1 ipaddress=172.30.3.1 mask=
      255.255.255.0
   ```

3. **Configure a unicast routing protocol.**

   Enable RIP over all interfaces.

   ```
   add ip rip int=eth0
   add ip rip int=eth1
   ```

4. **Configure IGMP.**

   Enable IGMP on the router for group management.

   ```
   enable ip igmp
   ```

   Enable IGMP on each interface, so that IGMP can find which multicast groups have hosts connected to the interfaces.

   ```
   enable ip igmp interface=eth0
   enable ip igmp interface=eth1
   ```

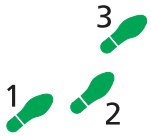
5. **Configure PIM.**

   Define PIM interfaces for the Ethernet interfaces.

   ```
   add pim interface=eth0
   add pim interface=eth1
   ```

   Enable PIM multicast routing.

   ```
   enable pim
   ```



Configure multicast routing using PIM Sparse Mode on Router C

1. **Set the system name.**

   Set a unique system name on the router.

   ```
   set sys name=C-pim
   ```

2. **Create a PPP interface.**

   Create a PPP interface 0 over synchronous port 0.

   ```
   create ppp=0 over=syn0
   ```

3. **Configure IP.**

   Enable IP on the router.

   ```
   enable ip
   ```

   Assign IP addresses to the PPP and Ethernet interfaces.

   ```
   add ip interface=ppp0 ipaddress=172.30.2.2
      mask= 255.255.255.0

   add ip interface=eth0 ipaddress=172.30.4.1
      mask= 255.255.255.0
   ```

4. **Configure a unicast routing protocol.**

   Enable RIP over all interfaces.

   ```
   add ip rip int=ppp0

   add ip rip int=eth0
   ```

5. **Configure IGMP.**

   Enable IGMP on the router for group management.

   ```
   enable ip igmp
   ```

   Enable IGMP on each interface, so that IGMP can find which multicast groups have hosts connected to the interfaces.

   ```
   enable ip igmp interface=ppp0

   enable ip igmp interface=eth0
   ```

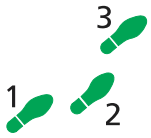
6. **Configure PIM.**

   Define PIM interfaces for the Ethernet and PPP interfaces.

   ```
   add pim interface=eth0

   add pim interface=ppp0
   ```

   Enable PIM multicast routing.

   ```
   enable pim
   ```



Confirm multicasting

When the three routers have been configured, RIP takes a few seconds to distribute the unicast routing information to all routers. The IP hosts connected to these interfaces can then send and receive multicasts.

1. **Test multicasting.**

   Test whether IP multicasting is successful by sending IP multicast data between hosts connected to each of the routers. Check that IGMP report and leave messages are correctly processed by having hosts leave and join groups.

2. **Check the multicast state.**

   To check each router, use the commands:

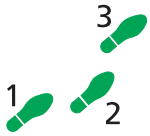
   ```
   show pim

   show ip igmp

   show ip route multicast
   ```

PIM Dense Mode

This example uses PIM Dense Mode for multicast routing between routers in the same topology as the PIM Sparse Mode example (Figure 17-4 on page 17-31). Multicast group management uses IGMP. The example assumes that each router starts from the default configuration.

Some interface and port types mentioned in this example may not be supported on your router. The interface and port types that are available vary depending on your product's model, and whether an expansion unit (PIC, NSM) is installed. For more information, see the AR400 Series Router Hardware Reference.

The configurations of Router A, B, and C are identical except for names and interfaces.



Configure multicast routing using PIM Dense Mode on Router A

1. **Set the system name for the router.**

```
set sys name=A-pim-dm
```

2. **Create a PPP interface.**

   Create a PPP interface 0 over synchronous port 0.

```
create ppp=0 over=syn0
```

3. **Configure IP.**

   Enable IP and assign IP addresses for the PPP and Ethernet interfaces on the router.

```
enable ip

add ip interface=eth0 ipaddress=172.30.1.1
   mask=255.255.255.0

add ip interface=ppp0 ipaddress=172.30.2.1
   mask=255.255.255.0
```

4. **Configure a unicast routing protocol.**

   Enable RIP over all interfaces.

```
add ip rip int=eth0

add ip rip int=ppp0
```

5. **Configure IGMP.**

   Enable IGMP on the router for group management.

```
enable ip igmp
```

   Enable IGMP on each interface, so that IGMP can find which multicast groups have hosts connected to the interfaces.

```
enable ip igmp interface=eth0

enable ip igmp interface=ppp0
```

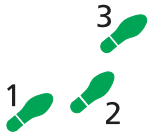
6. **Configure PIM.**

   Define PIM interfaces for the Ethernet and PPP interfaces.

```
add pim interface=eth0 mode=dense

add pim interface=ppp0 mode=dense
```

   Enable PIM multicast routing.

```
enable pim
```



Configure multicast routing using PIM Dense Mode on Router B

1. **Set the system name.**

   Set a unique system name on the router.

   ```
   set sys name=B-pim
   ```

2. **Configure IP.**

   Enable IP, and assign IP addresses to the Ethernet interfaces.

   ```
   enable ip

   add ip interface=eth0 ipaddress=172.30.1.2
      mask=255.255.255.0

   add ip interface=eth1 ipaddress=172.30.3.1
      mask=255.255.255.0
   ```

3. **Configure a unicast routing protocol.**

   Enable RIP over all interfaces.

   ```
   add ip rip int=eth0

   add ip rip int=eth1
   ```

4. **Configure IGMP.**

   Enable IGMP on the router for group management.

   ```
   enable ip igmp
   ```

   Enable IGMP on each interface, so that IGMP can find which multicast groups have hosts connected to the interfaces.

   ```
   enable ip igmp interface=eth0

   enable ip igmp interface=eth1
   ```

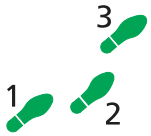
5. **Configure PIM.**

   Define PIM interfaces for the Ethernet interfaces.

   ```
   add pim interface=eth0 mode=dense

   add pim interface=eth1 mode=dense
   ```

   Enable PIM multicast routing.

   ```
   enable pim
   ```



Configure multicast routing using PIM Dense Mode on Router C

1. **Set the system name.**

   Set a unique system name on the router.

   ```
   set sys name=C-pim
   ```

2. **Create a PPP interface.**

   Create a PPP interface 0 over synchronous port 0.

   ```
   create ppp=0 over=syn0
   ```

3. **Configure IP.**

   Enable IP on the router.

   ```
   enable ip
   ```

   Assign IP addresses to the PPP and Ethernet interfaces.

   ```
   add ip interface=ppp0 ipaddress=172.30.2.2
      mask= 255.255.255.0

   add ip interface=eth0 ipaddress=172.30.4.1
      mask= 255.255.255.0
   ```

4. **Configure a unicast routing protocol.**

   Enable RIP over all interfaces.

   ```
   add ip rip int=ppp0

   add ip rip int=eth0
   ```

5. **Configure IGMP.**

   Enable IGMP on the router for group management.

   ```
   enable ip igmp
   ```

   Enable IGMP on each interface, so that IGMP can find which multicast groups have hosts connected to the interfaces.

   ```
   enable ip igmp interface=ppp0

   enable ip igmp interface=eth0
   ```

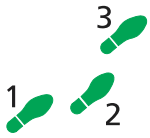
6. **Configure PIM.**

   Define PIM interfaces for the Ethernet and PPP interfaces.

   ```
   add pim interface=eth0 mode=dense

   add pim interface=ppp0 mode=dense
   ```

   Enable PIM multicast routing.

   ```
   enable pim
   ```



Confirm multicasting

When the three routers have been configured, RIP takes a few seconds to distribute the unicast routing information to all routers. Then the IP hosts connected to these interfaces can send and receive multicasts.

1. **Test multicasting.**

   Test whether IP multicasting is successful by sending IP multicast data between hosts connected to each of the routers. Check that IGMP report and leave messages are correctly processed by having hosts leave and join groups.

2. **Check the multicast state.**

   To check each router, use the commands:

   ```
   show pim

   show ip igmp

   show ip route multicast
   ```

Command Reference

This section describes the commands available on the router to configure IGMP for multicast group management, and the multicast routing protocols DVMRP (Distance Vector Multicast Routing Protocol), PIM-SM (Protocol Independent Multicast - Sparse Mode) and PIM-DM (Protocol Independent Multicast - Dense Mode).

Some interface and port types mentioned in this chapter may not be supported on your router. The interface and port types that are available vary depending on your product's model, and whether an expansion unit (PIC, NSM) is installed. For more information, see the AR400 Series Router Hardware Reference.

The shortest valid command is denoted by capital letters in the Syntax section. See "Conventions" on page xcv of Preface in the front of this manual for details of the conventions used to describe command syntax. See Appendix A, Messages for a complete list of error messages and their meanings.

add dvmrp interface

Syntax

```
ADD DVMrp INTerface=interface [DLC=1..1024] [METric=1..32]
    [TTLThreshold=1..255]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description

This command adds the specified interface to the DVMRP interface list, and starts DVMRP processes on the interface.

The **interface** parameter specifies the IP interface. The Layer 2 interface must already be configured. The IP interface must not already be assigned to the DVMRP module. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

The **metric** parameter specifies the metrics on a DVMRP interface. This metric is added to all routes that are learned via this interface. The default is 1.

The **ttlthreshold** parameter specifies the Time To Live (TTL) threshold on the DVMRP interface. Packets are forwarded out this interface when the TTL value in the packet is below this threshold. The default is 255.

Examples To add the interface eth0 as a DVMRP interface with a TTL threshold of 15, use the command:

```
add dvm int=eth0 ttlt=15
```

Related Commands
delete dvmrp interface
enable dvmrp
set dvmrp interface
show dvmrp

add igmpsnooping routeraddress

Syntax

```
ADD IGMPSNooping ROUTERAddress=ipaddr [,...]
```

where *ipaddr* is a reserved IP multicast address in dotted decimal notation, or a comma-separated list of reserved IP multicast addresses

Description This command adds reserved IP multicast addresses to the list of router multicast addresses. The IP address specified must be from 224.0.0.1 to 224.0.0.255. This command is valid when IGMP Snooping router mode is set to IP with the **set igmpsnooping routermode** command on page 17-60.

Examples To add addresses 224.0.0.25 and 224.0.0.86 to the router multicast address list, use the command:

```
add igmpsn routera=224.0.0.25,224.0.0.86
```

Related Commands
delete igmpsnooping routeraddress
set igmpsnooping routermode
.show igmpsnooping routeraddress

add pim bsrcandidate

Syntax

```
ADD PIM BSRCandidate [PREFerence=0..255]
   [HASHMASKLENGTH=0..32] [BSMinterval={10..15000|
   DEFault}][INTERFACE={local-interface|vlan-interface}]
```

Description This command configures the router to be a Bootstrap Router candidate.

The **preference** parameter specifies the preference for the router to become the bootstrap router. A higher number means a higher priority. The default is 1.

The **hashmasklength** parameter specifies the number of bits of the group number to use when selecting a rendezvous point (RP) candidate if this switch becomes the BSR. A higher number increases the spread of groups across RPs. The default is 30.

Note that previous release versions did not correctly support the PIM hash mask length option. As a result, the RP selection calculation differs between old and new release versions. If a network contains switches running a mixture of versions, this leads to incorrect forwarding behaviour. To avoid this issue, either ensure that all devices on the network correctly support the hash mask length option (recommended), or ensure that the following **both** hold:

- The hash mask length option on all BSR candidates is configured to 4 bits. This implies that all BSR candidates must be running a new release.
- All RP candidates use a common prefix of 224.0.0.0/240.0.0.0.

This will have the side effect of collapsing all groups to use a single PIM RP.

The **bsminterval** parameter can be used to specify the time period in seconds at which the router sends bootstrap messages when it is elected as the bootstrap router. The default is 60 seconds. This timer is now set with the **set pim** command by preference because it applies globally to the PIM-SM domain, but this parameter has been maintained to ensure backwards compatibility.

The **interface** parameter specifies an interface for the router to use when advertising itself as a candidate bootstrap router. The IP address of the of this interface will be advertised by the router. The interface supplied can be either a configured local interface or a configured VLAN interface. If the parameter is not specified the router will instead advertise its first active IP interface.

Examples

To add the router as a Bootstrap Router Candidate to a PIM domain, with a preference of 10 to become the bootstrap router in the domain and a hash mask length of 0, use the command:

```
add pim bsrcandidate preference=10 hasmasklength=0
```

Related Commands

delete pim bsrcandidate
enable pim
set pim bsrcandidate
show pim
show pim bsrcandidate

add pim interface

Syntax

```
ADD PIM INTerface=interface [DLCi=1..1024]
   [DRPriority=0..4294967295] [ELectby={DRPriority|
   IPaddress}] [HEllointerval={10..15000|DEFault|65535}]
   [MODe={Dense|Sparse}] [SRCapable={Yes|No}]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description This command adds the specified IP interface to the PIM interface list so that PIM multicast routing can operate on this interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlci** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

The **drpriority** parameter specifies the preference for the router to become the designated router (DR) on this interface when **electby**=**drpriority**. A higher value indicates a greater preference. The default is 1.

*For compatibility with previous versions, a DR priority of 65535 is treated as **electby=ipaddress** if **electby** has not been specified. If **electby=drpriority** is specified and **drpriority**=65535, then the DR priority is set to 65535.*

The **electby** parameter determines how the router elects the designated router for this interface. If **drpriority** is specified, the interface transmits its DR priority in its hello messages, which allows DR election by priority. If **ipaddress** is specified, the router does not transmit its DR priority, which causes election by IP address. The default is **drpriority**. Note that a router with **electby**=**drpriority** may still elect by IP address when it does not receive DR priority in any one of its neighbours' Hello messages. Election by DR priority is possible only when all routers on the interface supply their DR priority.

The **hellointerval** parameter specifies the interval at which the router sends Hello messages from this interface. Setting the **hellointerval** parameter to 65535 results in a Hello message being sent with a hold time of 65535, which means "infinity". A router receiving this router's Hello never expires this router as a PIM neighbour. This can be useful on point-to-point links. The default is 30 seconds.

The **mode** parameter specifies the PIM operating mode for the interface. The default is **sparse**.

All interfaces should have the same mode setting unless the router is a Multicast Border Router.

The **srcapable** parameter indicates whether this interface originates or processes State Refresh messages. The default is **no**. This parameter applies to dense mode interfaces.

Examples To add interface eth0 to the PIM-SM interface list, with a priority of 3 to become the designated router for the subnetwork, use the command:

```
add pim int=eth0 drp=3
```

Related Commands

delete pim interface
enable pim
reset pim interface
set pim interface
show pim
show pim interface

add pim rpcandidate

Syntax

```
ADD PIM RPCandidate[=rp-address] GROup=group-address
    [MASK=ipaddress] [PRIOrity=0..255]
    [ADVinterval={10..15000|DEFault}]
```

where:

- *group-address* is the IP address of the multicast group in dotted decimal notation.
- *ipaddress* is an IP address in dotted decimal notation.
- *rp-address* is an IP address in dotted decimal notation.

Description

This command configures the router to be a rendezvous point candidate for the specified multicast group(s). There is no limitation on the number of groups or group ranges.

The **rpcandidate** parameter, if specified with a value, is the IP address of the Rendezvous Point for the multicast group(s). This option can be used to create static RP mappings for networks in which the bootstrap mechanism cannot be used.

If the bootstrap mechanism is also running, a static RP mapping takes precedence.

The **group** parameter specifies the multicast group(s) to which the router is a rendezvous point candidate.

The **mask** parameter specifies the mask for the multicast group address specified in the **group** parameter. This is useful when configuring multiple multicast groups with a common Rendezvous Point (RP). The default mask is 255.255.255.255.

The **priority** parameter specifies the preference for the router to become the Rendezvous Point for the multicast group. A lower value indicates a higher priority. The default is 192. This parameter does not apply to static RP mappings.

*The router has the same values for **priority** for all multicast groups for which it is a rendezvous point candidate, so changing this router's priority to be the RP for one group changes it for all groups.*

The **advinterval** parameter can be used to specify the time period in seconds at which the router sends C-RP-Advertisements. The default is 60 seconds. This

timer is now set with the **set pim** command by preference because the router sends C-RP-Advertisements at the same rate for all groups for which it is an RP candidate, but this parameter has been maintained to ensure backwards compatibility. This parameter does not apply to static RP mappings.

The **interface** parameter specifies an interface for the router to use when advertising itself as the candidate rendezvous point for a multicast group. The IP address of the of this interface will be advertised by the router. The INTERFACE supplied can be either a configured local interface or a configured VLAN interface. If the parameter is not specified the router will advertise its first active IP interface instead.

Examples To configure the router to advertise that it is an RP candidate with a priority of 10 to become the RP for the multicast group with address 224.1.1.98, use the command:

```
add pim rpc gro=224.1.1.98 prio=10
```

Related Commands

delete pim rpcandidate
enable pim
set pim rpcandidate
show pim
show pim rpcandidate

delete dvmrp interface

Syntax `DELete DVMrp INTerface=`*`interface`* `[DLC=1..1024]`

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description This command deletes the specified interface from the DVMRP interface list, stops all DVMRP processes for the interface, and deletes all routing information generated by this interface.

The **interface** parameter specifies the IP interface. The IP interface must already be assigned to the DVMRP module. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

Examples To delete interface eth0 from the DVMRP interface list, use the command:

```
del dvm int=eth0
```

Related Commands
add dvmrp interface
disable dvmrp
reset dvmrp interface
show dvmrp

delete igmpsnooping routeraddress

Syntax
```
DELete IGMPSNooping ROUTERAddress=ipaddr [,...]
```

where *ipaddr* is a reserved IP multicast address in dotted decimal notation, or a comma-separated list of reserved IP multicast addresses

Description
This command deletes reserved IP multicast addresses from the list of router multicast addresses. The IP address specified must be from 224.0.0.1 to 224.0.0.255. This command is only valid if IGMP Snooping router mode is set to IP with the **set igmpsnooping routermode** command on page 17-60.

Examples
To remove addresses 224.0.0.25 and 224.0.0.86 from the router multicast address list, use the command:

```
del igmpsn routera=224.0.0.25,224.0.0.86
```

Related Commands
add igmpsnooping routeraddress
set igmpsnooping routermode
.show igmpsnooping routeraddress

delete pim bsrcandidate

Syntax
```
DELete PIM BSRCandidate
```

Description
This command stops the router from acting as a bootstrap router candidate in the PIM-SM domain.

Examples
To stop the router from acting as a bootstrap router candidate, use the command:

```
del pim bsrc
```

Related Commands
add pim bsrcandidate
disable pim
show pim
show pim bsrcandidate

delete pim interface

Syntax

```
DELete PIM INTerface=interface [DLCi=1..1024]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description

This command deletes the specified interface from the PIM interface list on the router, stops all PIM processes on the interface, and deletes all routing information generated by the interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show pim interface** command on page 17-84. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlci** parameter specifies the DLCI of a Frame Relay interface. The **dlci** parameter is valid for Frame Relay interfaces.

Examples

To delete interface eth0 from PIM interface list, use the command:

```
del pim int=eth0
```

Related Commands

add pim interface
disable pim
show pim
show pim interface

delete pim rpcandidate

Syntax

```
DELete PIM RPCandidate[=rp-address] GROup=group-address
    [MASK=ipaddress]
```

where:

- *group-address* is the IP address of the multicast group in dotted decimal notation.
- *ipaddress* is an IP address in dotted decimal notation.
- *rp-address* is an IP address in dotted decimal notation.

Description

This command deconfigures the router from acting as a Rendezvous Point candidate for the specified multicast group(s).

The **rpcandidate** parameter is the IP address of the Rendezvous Point for the multicast group(s) when it is specified with a value. This option can be used to remove a static RP mapping.

The **mask** parameter specifies the mask for the multicast group address specified with the **group** parameter. This is useful when deconfiguring multiple multicast groups with a common Rendezvous Point (RP). The default mask is 255.255.255.255.

Examples To stop the router from advertising itself as an RP candidate for multicast group 224.1.1.98, use the command:

```
del pim rpc gro=224.1.1.98
```

Related Commands
add pim rpcandidate
disable pim
show pim
show pim rpcandidate

disable dvmrp

Syntax
```
DISable DVMrp
```

Description This command disables the DVMRP module thereby stopping the DVMRP routing process. Other DVMRP configuration remains intact. By default, DVMRP is disabled when the router is started.

Example To disable DVMRP, use the command:

```
dis dvm
```

Related Commands
delete dvmrp interface
enable dvmrp
reset dvmrp interface
show dvmrp

disable dvmrp debug

Syntax
```
DISable DVMrp DEBug={ALL|GRAft|PRObe|PRUne|REPort}[,...]
   INTerface=interface [DLC=1..1024]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a

logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description This command disables the debugging option for the specified DVMRP interface. The option must currently be enabled. A list of options separated by commas may be specified to disable more than one debugging option at a time. By default all DVMRP debugging is disabled. To see a list of current interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show pim interface** command on page 17-84.

The **debug** parameter specifies debugging options that are disabled. The value of this parameter is a single item or a comma-separated list of items. The debugging that is disabled by each of the options is shown in Table 17-1 on page 17-51.

The **interface** parameter specifies the DVMRP interface already assigned and configured, which is to be debugged. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show dvmrp interface** command on page 17-69.

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

Example To disable all DVMRP debugging on interface eth0, use the command:

```
dis dvm deb int=eth0
```

Related Commands
enable dvmrp debug
show dvmrp

disable igmpsnooping

Syntax `DISable IGMPSNooping`

Description This command disables IGMP snooping on the router. IGMP snooping is enabled by default. Note that multicast packets flood the VLAN when IGMP snooping is disabled.

Disabling IGMP snooping may be useful when filters are used extensively because IGMP snooping uses a Layer 3 filter. When IGMP snooping is disabled, this filter becomes available. *IGMP snooping is independent of IGMP. IGMP is disabled by default.*

Examples To disable IGMP snooping, use the command:

```
dis igmpsn
```

Related Commands
disable ip igmp interface
enable igmpsnooping
enable ip igmp
enable ip igmp interface
show ip igmp

disable ip igmp

Syntax
```
DISable IP IGMP
```

Description
This command disables IGMP on the router so that multicast routing stops immediately. IGMP is disabled by default.

Examples
To disable the IGMP module, use the command:

```
dis ip igmp
```

Related Commands
disable ip igmp interface
enable ip igmp
show ip igmp

disable ip igmp allgroup

Syntax
```
DISable IP IGMP ALLGroup=[port-list|ALL]
```

where:

- *port-list* is a port number, a range of port numbers (specified as n-m), or a comma separated list of port numbers and/or ranges. Port numbers start at 1 and end at m, where m is the highest numbered Ethernet switch port, including uplink ports.

Description
This command disables the specified port or ports from acting as a router port. Once disabled, the port will no longer receive MARL entries when the device receives an IGMP report, query, or multicast data over any other port.

Example
To prevent ports 1, 5, and 7 from acting as an all-group entry, use the command:

```
dis ip igmp allg=1,5,7
```

Related Commands
enable ip igmp allgroup

disable ip igmp debug

Syntax `DISable IP IGMP DEBug`

Description This command disables all IGMP debugging messages and resets the **destination** and **sourceipaddress** parameters set with the **enable ip igmp debug** command to **all**. Debugging is disabled by default.

Examples To disable all IGMP debugging messages and reset the IGMP debug message filters for all, use the command:

```
dis ip igmp deb
```

Related Commands **show ip igmp debug**

disable ip igmp interface

Syntax `DISable IP IGMP INTerface=`*`interface`* `[DLC=1..1024]`

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description This command disables IGMP on the specified interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces.

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the data link circuit identifier of a Frame Relay interface.

Examples To disable IGMP on interface eth0, use the command:

```
dis ip igmp int=eth0
```

Related Commands **disable ip igmp**
enable ip igmp interface
show ip igmp

disable pim

Syntax

```
DISable PIM
```

Description

This command disables PIM on the router. PIM multicast routing stops but PIM configurations remain intact. PIM is disabled by default.

Examples

To disable PIM on the router, use the command:

```
dis pim
```

Related Commands

delete pim bsrcandidate
delete pim interface
delete pim rpcandidate
enable pim
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

disable pim debug

Syntax

```
DISable PIM DEBug={ALL|ASSERT|BSR|C-RP-ADV|GRAft|HELLO|
   JOInt|REGister|STATerefresh}[,...]
```

Description

This command disables the debugging option. The option must currently be enabled. PIM debugging is disabled by default.

The **debug** parameter specifies which debugging options are to be disabled. The value of this parameter is a single option or a comma-separated list of options. The debugging that results from each of the options is shown in Table 17-2 on page 17-55.

Examples

To disable all PIM debugging, use the command:

```
dis pim deb=all
```

Related Commands

enable pim debug
show pim debug

enable dvmrp

Syntax

```
ENAble DVMrp
```

Description

This command enables DVMRP routing, and activates any existing DVMRP configuration. By default DVMRP is disabled when the router is started.

Example

To enable DVMRP, use the command:

```
ena dvm
```

Related Commands

add dvmrp interface
disable dvmrp
set dvmrp interface
show dvmrp debug

enable dvmrp debug

Syntax

```
ENAble DVMrp DEBug={ALL|GRAft|PRObe|PRUne|REPort}[,...]
   INTerface=interface [DLC=1..1024]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from. If a logical interface is not specified, 0 is assumed.

Description

This command enables debugging options for the specified DVMRP interface. Debugging may or may not be enabled already. Debugging information is sent to the port or Telnet session from which the command is entered. By default, DVMRP debugging is disabled.

The **debug** parameter specifies the debugging options that are enabled. The value of this parameter is a single option or a comma-separated list of options. The debugging that results from each of the options is shown in Table 17-1 on page 17-51.

Table 17-1: DVMRP debugging options.

Parameter	Meaning
all	All debug options
graft	DVMRP Graft and Graft Ack packets
probe	DVMRP Router Probe packets
prune	DVMRP Prune packets
report	DVMRP Route Report packets

The **interface** parameter enables DVMRP debugging on the specified interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show dvmrp interface** command on page 17-69.

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

Example To enable debugging of DVMRP prune and graft messages on the eth0 interface, use the command:

```
ena dvm deb=gra,pru int=eth0
```

Related Commands
disable dvmrp debug
show dvmrp

enable igmpsnooping

Syntax

```
ENAble IGMPSNooping
```

Description This command enables IGMP snooping on the router. IGMP snooping is enabled by default. IGMP snooping can be enabled only when a free Layer 3 filter is available.

IGMP snooping is independent of IGMP. IGMP is disabled by default.

Examples To enable IGMP snooping, use the command:

```
ena igmpsn
```

Related Commands
disable igmpsnooping
disable ip igmp
disable ip igmp interface
enable ip igmp
enable ip igmp interface
show ip igmp

enable ip igmp

Syntax `ENAble IP IGMP`

Description This command enables IGMP on the router. IGMP is disabled by default.

Examples To enable IGMP, use the command.

```
ena ip igmp
```

Related Commands
disable ip igmp
enable ip igmp interface
show ip igmp

enable ip igmp allgroup

Syntax `ENAble IP IGMP ALLGroup=[`*`port-list`*`|ALL]`

where:

- *port-list* is a port number, a range of port numbers (specified as n-m), or a comma separated list of port numbers and/or ranges. Port numbers start at 1 and end at m, where m is the highest numbered Ethernet switch port, including uplink ports.

Description This command enables the specified port or ports to act as a router port. All ports are allowed to be a router port by default, so this command is used to re-enable a port as a router port if it has previously been disabled with the **disable ip igmp allgroup** command.

Example To enable ports 1, 5, and 7 to act as an all-group entry, use the command:

```
ena ip igmp allg=1,5,7
```

Related Commands
disable ip igmp allgroup
show ip igmp

enable ip igmp debug

Syntax `ENAble IP IGMP DEBug [DEStination={ALL|`*`ipaddress`*`}] [SOURCEipaddress={ALL|`*`ipaddress2`*`}]`

where:

- *ipaddress* is an IGMP group destination address.
- *ipaddress2* is the IP address of a host that responds to IGMP queries.

Description This command enables IGMP debugging of destination and source IP addresses. Debugging is disabled by default.

The **destination** parameter specifies the destination multicast group address for debugging. The default is **all**.

The **sourceipaddress** parameter specifies the host IP address responding to IGMP queries. The default is **all**.

If **destination** and **sourceipaddress** are both specified, debug messages that match both parameters are displayed. Some debug messages are displayed before the packet is fully decoded, and are unable to be filtered.

Examples To enable debugging information relating to IGMP host 10.41.0.22, use the command:

```
ena ip igmp deb SOURCEIPADDRESS=10.41.0.22
```

To show all IGMP debug messages, use the command:

```
ena ip igmp deb
```

Related Commands **show ip igmp debug**

enable ip igmp interface

Syntax `ENAble IP IGMP INTerface=`*`interface`* `[DLC=1..1024]`

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface number is not specified, 0 is assumed.

Description This command enables IGMP to run on the specified interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces.

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

Examples To enable IGMP on eth0 interface, use the command:

```
ena ip igmp int=eth0
```

Related Commands **disable ip igmp interface**
enable ip igmp
show ip igmp

enable pim

Syntax

```
ENAble PIM
```

Description

This command enables PIM routing on the router. PIM is disabled by default. Any existing PIM configuration is activated after this command has been entered.

Examples

To enable PIM routing, use the command:

```
ena pim
```

Related Commands

add pim bsrcandidate
add pim interface
add pim rpcandidate
disable pim
set pim interface
set pim
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

enable pim debug

Syntax

```
ENAble PIM DEBug={ALL|ASSERT|BSR|C-RP-ADV|GRAft|HELLO|
   JOInt|REGister|STATerefresh}[,...]
```

Description

This command enables debugging options. Debugging may or may not be enabled already. Debugging information is sent to the port or Telnet session from which the command was entered. All PIM debugging is disabled by default.

The **debug** parameter specifies which debugging options are to be enabled. The value of this parameter is a single option or a comma-separated list of options. The options and debugging results are shown in Table 17-2 on page 17-55.

Table 17-2: PIM debugging options .

Parameter	Meaning
ALL	All debug options.
ASSERT	PIM Assert packets
BSR	PIM Bootstrap packets (Sparse Mode only)
C-RP-ADV	PIM Candidate-RP-Advertisement (Sparse Mode only)

Table 17-2: PIM debugging options (continued).

Parameter	Meaning
GRAFT	PIM Graft packets (Dense Mode only)
HELLO	PIM Hello packets
JOIN	PIM Join/Prune packets
REGISTER	PIM Register and Register Stop packets (Sparse Mode only)
STATEREFRESH	PIM State Refresh packets (Dense Mode only)

Examples To enable debugging of PIM Hello and Join/Prune messages, use the command:

```
ena pim deb=hello,joi
```

Related Commands **disable pim debug**

purge dvmrp

Syntax `PURge DVMrp`

Description This command disables DVMRP, purges all configuration information relating to the DVMRP multicast routing module, and reinitialises the data structures used by the module. It should be used when first setting up the DVMRP module or when a major change is required.

All current DVMRP configuration information will be lost. Use with extreme caution!

Related Commands
delete dvmrp interface
disable dvmrp
reset dvmrp interface
set dvmrp interface
show dvmrp

purge pim

Syntax `PURge PIM`

Description This command purges all configuration information relating to the PIM multicast routing module, and reinitialises the data structures used by the module. It also stops the current PIM operation. It should be used when first setting up the PIM module or when a major change is required.

All current PIM configuration information will be lost. Use with extreme caution!

Related Commands

delete pim bsrcandidate
delete pim interface
delete pim rpcandidate
disable pim
disable pim debug
reset pim interface
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

reset dvmrp interface

Syntax

```
RESET DVMrp INTerface=interface [DLC=1..1024]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description

This command resets all DVMRP processes, timers and route information for the specified interface. This effectively restarts all DVMRP processes for this interface as if the interface has just been added to the DVMRP interface list.

The **interface** parameter specifies the IP interface. The IP interface must already have been added to the DVMRP module. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show dvmrp interface** command on page 17-69. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

Examples

To reset interface eth0 in the DVMRP interface list, use the command:

```
reset dvm int=eth0
```

Related Commands

delete dvmrp interface
disable dvmrp
set dvmrp interface
show dvmrp

reset pim interface

Syntax

```
RESET PIM INTerface=interface [DLC=1..1024]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description

This command resets all timers, route information, and counters associated with the specified interface, and restarts all PIM processes for this interface as if this interface has just been added to PIM interface list. It also disables any enabled PIM debugging on the interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show pim interface** command on page 17-84. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

Examples

To reset the ppp0 interface, use the command:

```
reset pim int=ppp0
```

Related Commands

set pim interface
set pim
show pim

set dvmrp interface

Syntax

```
SET DVMrp INTerface=interface [DLC=1..1024] [METric=1..32]
   [TTLThreshold=1..255]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description

This command modifies the configuration of a DVMRP interface. When any parameter is modified, the DVMRP processes and any routing information generated by this interface is reset.

The **interface** parameter specifies the IP interface. The IP interface must already be assigned to the DVMRP module. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show dvmrp interface** command on page 17-69. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlc** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

The **metric** parameter specifies the metrics on a DVMRP interface. This metric is added to all routes that are learned via this interface. The default is 1.

The **ttlthreshold** parameter specifies the Time To Live threshold on a DVMRP interface in seconds. Packets are forwarded from this interface when the TTL in the packet is below this threshold. The default is 255.

Examples To change the metric for DVMRP interface eth0 to 4, use the command:

```
set dvm int=eth0 met=4
```

Related Commands

reset dvmrp interface
show dvmrp

set ip igmp

Syntax

```
SET IP IGMP [LMQi=1..255] [LMQC=1..5]
   [QUEryinterval=1..65535] [QUERYREsponseinterval=1..255]
   [ROBustness=1..5] [TIMEOut=1..65535]
```

Description This command sets operational timers and thresholds for IGMP.

The defaults for these timers suit most networks. Changing them to inappropriate values can cause IGMP to function in undesirable ways. System administrators should change timer values based on a sound understanding of their interaction with other devices in the network.

The **lmqi** parameter specifies the Last Member Query Interval (in 1/10 secs), which is the Max Response Time inserted into Group-Specific Queries sent in response to Leave Group messages. It is also the amount of time between Group-Specific Query messages. The default is 10 (1 second).

The **lmqc** parameter specifies the Last Member Query Count, which is the number of Group-Specific Queries sent before the router assumes there are no local members. The default is the same as ROBUSTNESS value.

The **queryinterval** parameter specifies the seconds of the interval between IGMP Host Membership Queries if this router is elected the designated router

for the LAN. If the router is not the IGMP designated router, it ignores this parameter. The default is 125.

The **queryresponseinterval** parameter specifies the Max Response Time (in 1/10 secs) inserted into the periodic General Queries. The default is 100 (10 seconds).

The **robustness** parameter specifies the Robustness Variable that allows tuning for the expected packet loss on a subnet. If a subnet is expected to be lossy, the Robustness Variable may be increased. IGMP is robust to (Robustness Variable-1) packet losses. The Robustness Variable *must not* be zero and *should not* be one. The default is 2.

The **timeout** parameter specifies the longest interval in seconds that a group remains in the local multicast group database without the router (designated router or not) receiving a Host Membership Report for this multicast group. This **timeout** parameter is used by all IGMP routers to maintain their group membership databases. The default is 270. If a value is specified for **queryinterval** without specifying a value for **timeout**, **timeout** is calculated as 2*(**queryinterval** + 10). The added 10 seconds is the variation that hosts use when sending Host Membership Reports. When a timeout interval is specified, it overrides a calculated value.

Examples To set the IGMP query interval to 180s (3 minutes), use the command:

```
set ip igmp que=180
```

Related Commands **enable ip igmp interface**
show ip igmp

set igmpsnooping routermode

Syntax

```
SET IGMPSNooping ROUTERMode=[ALL|DEFault|IP|
   MULTICAstrouter|NONE]
```

Description This command sets the mode of operation for IGMP Snooping.

The **all** parameter specifies that all reserved multicast addresses (i.e. 224.0.0.1 to 224.0.0.255) are treated as router multicast addresses.

The **default** parameter specifies the following multicast addresses:

Table 17-3: Default Multicast Addresses

Router Type	Multicast Address
IGMP Query	224.0.0.1
All routers on this subnet	224.0.0.2
DVMRP Routers	224.0.0.4
All OSPFIGP routers	224.0.0.5
OSPFIGP designated routers	224.0.0.6
RIP2 routers	224.0.0.9
All PIM routers	224.0.0.13
All CBT routers	224.0.0.15

The **ip** parameter specifies that addresses treated as router multicast addresses are specified by the user with the **add igmpsnooping routeraddress** command on page 17-39 and the **delete igmpsnooping routeraddress** command on page 17-44. When in this mode, the router retains previous addresses already specified.

The **multicastrouter** parameter specifies that the following addresses are treated as router multicast addresses:

- DVMRP Routers, 224.0.0.4
- All PIM routers, 224.0.0.13

The **none** parameter specifies that the router does not create router ports.

Examples To allow the router to treat all reserved multicast addresses as router multicast addresses, use the command:

```
set igmpsn routerm=all
```

Related Commands

add igmpsnooping routeraddress
delete igmpsnooping routeraddress
.show igmpsnooping routeraddress

set pim

Syntax

```
SET PIM [ADVinterval={10..15000|DEFault}]
   [BSMinterval={10..15000|DEFault}]
   [JPInterval={1..65535|DEFault}]
   [KEEPalivetime={10..65535|DEFault}]
   [PRObetime={1..65535|DEFAULT}]
   [PRUNEholdtime={1..65535|DEFault}]
   [SOURCealivetime={10..65535|DEFault}]
   [SRINterval={10..255|DEFault}]
   [SUPPressiontime={1..65535|DEFault}]
```

Description This command sets the timers for PIM operations.

The defaults for these timers suit most networks. Changing them to inappropriate values may cause PIM to function in undesirable ways. System administrators should change these timer values based on a sound understanding of their interaction with other devices in the network.

The **advinterval** parameter specifies the seconds of the interval at which the router sends C-RP-Advertisements. The default is 60 seconds. This timer applies to PIM-SM only.

The **bsminterval** parameter specifies the seconds of the interval at which the router sends bootstrap messages when it is the bootstrap router in the domain. The default is 60 seconds. This timer applies to PIM-SM only.

The **jpinterval** parameter specifies the Upstream Join Timer in seconds. This is the interval at which PIM Join/Prune messages are sent. For proper operation, a maximum value of 18000 seconds is recommended. The default is 60 seconds.

The **keepalivetime** parameter specifies the seconds that the Join state for a particular Source and Group pair is maintained in the absence of data for that pair. The default is 210 seconds.

The **probetime** interval specifies the Register Probe Time in seconds. This is the time the DR waits for another Register Stop message after sending a Null Register message to the RP. If it does not receive a Register Stop message in this time, it resumes registering data packets to the RP. The default is 5 seconds. This timer applies to PIM-SM only.

The **pruneholdtime** parameter specifies the seconds that the prune state is maintained. This time is used in Prune messages to let upstream neighbours know how long to hold the prune state. It is also used as the Prune Limit Timer for suppressing prunes if a Prune message has already been sent. The default is 60 seconds. This timer applies to PIM-DM only.

The **sourcealivetime** parameter specifies the seconds that a router acting as a State Refresh Originator is active in the absence of data packets from the source. The default is 210 seconds. This timer applies to PIM-DM only.

The **srinterval** parameter specifies the seconds of the interval at which this router sends State Refresh Messages, if it is configured to be State Refresh Capable, and becomes a State Refresh Originator (in general, this means having a directly connected source). The default is 60 seconds. This timer applies to PIM-DM only.

The **suppressiontime** parameter specifies the Register Suppression Time. This determines the interval at which the sender's DR sends Null Register messages to the group's RP to tell it to send another Register Stop message if it still does not need the data to be registered and sent to it. The default is 60 seconds. This timer applies to PIM-SM only.

Examples

To set the Join/Prune Message interval to 90 seconds, use the command:

```
set pim jpi=90
```

Related Commands

enable pim
set pim interface
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

set pim log

This command applies to PIM-SM only.

Syntax

```
SET PIM LOG=[NONE|STAtus|ERRor|ALL] [TRap=[NONE|STAtus|
   ERRor|ALL]]
```

Description

This command sets the type of logging for status PIM log messages, error messages and/or sending of SNMP traps for certain error conditions.

The LOG parameter specifies whether status, error, or all log messages should be generated. The default is status.

The TRAP parameter specifies whether status, error, or all traps should be generated.

Related Commands

show pim

set pim bsrcandidate

Syntax

```
SET PIM BSRCandidate [HASHMASKLENGTH=0..32]
   [PREFerence=0..255]
```

Description

This command sets the router's Bootstrap Router Candidate preference. The **hashmasklength** parameter specifies the number of bits of the group number to use when selecting a rendezvous point (RP) candidate if this switch becomes the BSR. A higher number increases the spread of groups across RPs. The default is 30.

Note that previous release versions did not correctly support the PIM hash mask length option. As a result, the RP selection calculation differs between old and new release versions. If a network contains switches running a mixture of versions, this leads to incorrect forwarding behaviour. To avoid this issue, either ensure that all devices on the network correctly support the hash mask length option (recommended), or ensure that the following **both** hold:

- The hash mask length option on all BSR candidates is configured to 4 bits. This implies that all BSR candidates must be running a new release.
- All RP candidates use a common prefix of 224.0.0.0/240.0.0.0.

This will have the side effect of collapsing all groups to use a single PIM RP.

The **preference** parameter specifies the preference for this router to become the bootstrap router for the PIM-SM domain. A higher value indicates a greater preference.

Examples

To change the router's candidate BSR preference to 100 and a hash mask length of 0, use the command:

```
set pim bsrc pref=100 hasmasklength=0
```

Related Commands

add pim bsrcandidate
delete pim bsrcandidate
show pim bsrcandidate

set pim interface

Syntax

```
SET PIM INTerface=interface [DLCi=1..1024]
   [DRPriority=0..4294967295] [ELectby={DRPriority|
   IPaddress}] [HEllointerval={10..15000|DEFault|65535}]
   [MODe={Dense|Sparse}] [SRCapable={Yes|No}]
```

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description

This command sets parameters for the specified PIM interface. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

To see a list of current valid interfaces, use the **show interface** command on page 7-66 of Chapter 7, Interfaces, or the **show pim interface** command on page 17-84. Note that multihomed interfaces must specify the logical interface number (e.g. ppp1-1).

The **dlci** parameter is valid for Frame Relay interfaces, and specifies the DLCI of a Frame Relay interface.

The **drpriority** parameter specifies the preference for the router to become the designated router (DR) on this interface when **electby**=**drpriority**. A higher value indicates a greater preference. The default is 1.

The **electby** parameter determines how the router elects the designated router for this interface. If **drpriority** is specified, the interface transmits its drpriority in its hello messages, which allows DR election by priority. If **ipaddress** is specified, the router does not transmit its DR priority, which causes election by IP address. The default is **drpriority**. Note that a router with **electby**=**drpriority** may still elect by IP address when it does not receive DR priority in any one of its neighbours' hello messages. Election by DR priority is possible only when all routers on the interface supply their DR priority.

The **hellointerval** parameter specifies the interval at which the router sends Hello messages from this interface. Setting the **hellointerval** parameter to 65535 results in a Hello message being sent with a hold time of 65535, which means "infinity". A router receiving this router's Hello never expires this router as a PIM neighbour. This can be useful on point-to-point links. The default is 30 seconds.

The **mode** parameter specifies the PIM operating mode for the interface. The default is **sparse**.

All interfaces should have the same mode setting unless the router is a Multicast Border Router.

The **srcapable** parameter indicates if this interface is able to originate or process State Refresh messages. The default is **no**. This parameter applies to Dense Mode interfaces only.

Examples

To set the designated router priority for the interface eth0 to 100, use the command:

```
set pim int=eth0 drp=100
```

Related Commands

add pim interface
delete pim interface
enable pim
reset pim interface
show pim
show pim interface

set pim rpcandidate

Syntax

```
SET PIM RPCandidate GROup=group-address [MASK=ipaddress]
   [PRIOrity=0..255][INTERFACE={local-interface|vlan-
   interface}]
```

where:

- *group-address* is the IP address of the multicast group in dotted decimal notation
- *ipaddress* is an IP address in dotted decimal notation

Description

This command sets the Rendezvous Point Candidate priority for the specified multicast group(s).

The **group** parameter specifies the multicast group or groups to which the router is a rendezvous point candidate.

The **mask** parameter specifies the mask for the multicast group address specified in the **group** parameter. This is useful when configuring multiple multicast groups with a common Rendezvous Point (RP). The default mask is 255.255.255.255. The mask for a group cannot be modified.

The **priority** parameter specifies the preference for the router to become the Rendezvous Point for the multicast group. A lower value indicates a higher priority. The default is 192.

The **interface** parameter specifies an interface for the router to use when advertising itself as the candidate rendezvous point for a multicast group. The IP address of the of this interface will be advertised by the router. The INTERFACE supplied can be either a configured local interface or a configured

VLAN interface. If the parameter is not specified the router will advertise its first active IP interface instead.

*The router has the same values for **priority** for all multicast groups for which it is a rendezvous point candidate, so changing this router's priority to be the RP for one group changes it for all groups.*

Examples To change the router's RP candidate priority to 10 for the multicast group with address 224.1.1.98, use the command:

```
set pim rpc gro=224.1.1.98 prio=10
```

Related Commands

add pim rpcandidate
delete pim rpcandidate
show pim rpcandidate
show pim rpset

show dvmrp

Syntax `SHow DVMrp`

Description This command displays detailed information about DVMRP routing status on the router, and is the equivalent of specifying the following commands:

```
show dvmrp counters
show dvmrp debug
show dvmrp interface
show dvmrp neighbour
show dvmrp route
```

Examples To display detailed information about DVMRP, use the command:

```
sh dvm
```

Related Commands

add dvmrp interface
delete dvmrp interface
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface
set dvmrp interface

show dvmrp counters

Syntax `SHow DVMrp COUnters`

Description This command displays all DVMRP counters (see Figure 17-5 on page 17-67, Table 17-4 on page 17-67).

Figure 17-5: Example output from the **show dvmrp counters** command.

```
DVMRP Interface Counters
Interface: eth0
            Rcv Pkts     Rcv Bad Pkts   Send Pkts
-------------------------------------------------------------
Probe       0000000280   0000000000     0000000281
Report      0000000042   0000000000     0000000042
Prune       0000000000   0000000000     0000000000
Graft       0000000000   0000000000     0000000000
GraftAck    0000000000   0000000000     0000000000
Total       0000000322   0000000000     0000000323
-------------------------------------------------------------
Interface: eth1
            Rcv Pkts     Rcv Bad Pkts   Send Pkts
-------------------------------------------------------------
Probe       0000000000   0000000000     0000000001
Report      0000000000   0000000000     0000000000
Prune       0000000000   0000000000     0000000000
Graft       0000000000   0000000000     0000000000
GraftAck    0000000000   0000000000     0000000000
Total       0000000000   0000000000     0000000001
-------------------------------------------------------------
```

Table 17-4: Parameters in the output of the **show dvmrp counters** command.

Parameter	Meaning
Interface	IP interfaces running DVMRP processes.
Rcv Pkts	The number of packets receive in the interface.
Rcv Bad Pkts	The number of DVMRP messages received on the interface by the DVMRP process which were subsequently discarded as invalid (e.g. invalid packet format, or a route report from an unknown neighbour).
Sent Pkts	The number of packets sent to the interface.

Examples To display DVMRP counters, use the command:

```
sh dvm cou
```

Related Commands

add dvmrp interface
delete dvmrp interface
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface
set dvmrp interface

show dvmrp debug

Syntax `SHow DVMrp DEBug`

Description This command displays the interface debugging options (see Figure 17-6 on page 17-68, Table 17-5 on page 17-68).

Figure 17-6: Example output from the **show dvmrp debug** command.

```
DVMRP Debug
Interface    Debug Options
-------------------------------------------------------------
eth0         None
eth1         None
-------------------------------------------------------------
```

Table 17-5: Parameters in the output of the **show dvmrp debug** command.

Parameter	Meaning
Interface	IP interfaces running DVMRP processes.
Debug Options	The debugging options enabled on the specified interface.

Examples To display the debug options that are enabled, use the command:

```
sh dvm deb
```

Related Commands
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface

show dvmrp forwarding

Syntax `SHow DVMrp FORwarding`

Description This command displays the DVMRP forwarding table (see Figure 17-7 on page 17-68, Table 17-6 on page 17-69).

Figure 17-7: Example output from the **show dvmrp forwarding** command.

```
DVMRP forwarding table
Source Address      Source Mask       Group         In Port    Pruned Up
  Forwarding Ports<DS|Prune|DR|LocalHost>
-------------------------------------------------------------------------------
128.7.6.4         255.255.255.255  224.5.5.5        Eth0     No
  Eth1<1|0|Yes|No> Fr0-1<0|0|Yes|Yes>

-------------------------------------------------------------------------------
```

Table 17-6: Parameters in the output of the **show dvmrp forwarding** command.

Parameter	Meaning
Interface	IP interfaces running DVMRP processes.
Source Address	The IP address of sources for multicast packets.
Source Mask	The mask for the Source Address.
Group	The IP address of the multicast group.
In Port	The interface for incoming packets.
Forwarding Ports	The interface from which packets are forwarded.

Examples To display the DVMRP forwarding table, use the command:

```
sh dvm for
```

Related Commands

add dvmrp interface
delete dvmrp interface
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface
set dvmrp interface

show dvmrp interface

Syntax `SHow DVMrp INTerface`

Description This command displays the DVMRP interface list (see Figure 17-8 on page 17-69, Table 17-7 on page 17-70). Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

Figure 17-8: Example output from the **show dvmrp interface** command.

```
DVMRP Interface Table
Interface    Metric    TTL Threshold
-------------------------------------------------------------
eth0         001       00255
eth1         001       00255
-------------------------------------------------------------
```

Table 17-7: Parameters in the output of the **show dvmrp interface** command.

Parameter	Meaning
Interface	IP interfaces running DVMRP processes.
Metric	The metrics on a DVMRP interface. This metric is added to all routes that are learned via this interface
TTL Threshold	The Time To Live threshold on a DVMRP interface. A packet is not forwarded from this interface unless the TTL value in the packet is below this threshold.

Examples To display a list of all DVMRP interfaces, use the command:

```
sh dvm int
```

Related Commands
add dvmrp interface
delete dvmrp interface
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface
set dvmrp interface

show dvmrp neighbour

Syntax

```
SHow DVMrp NEIghbour
```

Description This command displays the contents of the DVMRP Neighbour Table (see Figure 17-9 on page 17-70, Table 17-8 on page 17-70).

Figure 17-9: Example output from the **show dvmrp neighbour** command.

```
DVMRP Neighbour Table
Interface   IP Address        Two Way
-------------------------------------------------------------
eth0        192.168.196.2     No
-------------------------------------------------------------
```

Table 17-8: Parameters in the output of the **show dvmrp neighbour** command.

Parameter	Meaning
Interface	IP interfaces running DVMRP processes.
IP address	The IP address of the DVMRP neighbour.
Two Way	Whether the DVMRP neighbour has also recognised this router as its neighbour.

Examples To display the DVMRP neighbour table, use the command:

```
sh dvm nei
```

Related Commands

add dvmrp interface
delete dvmrp interface
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface
set dvmrp interface

show dvmrp route

Syntax `SHow DVMrp ROUte`

Description This command displays the internal DVMRP routing table (see Figure 17-10 on page 17-71, Table 17-9 on page 17-71).

Figure 17-10: Example output from the **show dvmrp route** command.

```
DVMRP Routing Table

Source Address    Source Mask      Metric  Next Hop           Hold Down
 Designated Route
  Dependent Neighbours
------------------------------------------------------------------------------
128.1.0.0         255.255.0.0      8       PPP0->137.39.3.93  No
 Eth0->me  Eth1->me
  Eth0->138.5.8.1  Eth1->139.4.4.0

128.2.0.0         155.255.0.0      3       Eth0->128.7.5.2    No
 Eth1->139.4.4.0

------------------------------------------------------------------------------
```

Table 17-9: Parameters in the output of the **show dvmrp route** command.

Parameter	Meaning
Source Address	The IP address of sources for multicast packets.
Source Mask	The mask for the Source Address.
Metric	The administrative metric assigned to this route.
Next Hop	The IP address of the next hop router to get to the source.
Hold Down	Whether the route is held down. If yes, the route is no longer available but has not been deleted from the multicast routing table.
Designated Router	The interface and IP address of the designated router for this interface, or "me" if this router is the designated router for the interface.
Dependent Neighbours	The interface and IP address of DVMRP neighbours dependent on this router.

Examples To display the DVMRP routing table, use the command:

```
sh dvm rou
```

Related Commands

add dvmrp interface
delete dvmrp interface
disable dvmrp
disable dvmrp debug
enable dvmrp
enable dvmrp debug
reset dvmrp interface
set dvmrp interface

.show igmpsnooping routeaddress

Syntax

```
SHow IGMPSNooping ROUTERAddress
```

Description This command displays the current list of configured IP multicast router addresses configured on the router.

Figure 17-11: Example output from the **show igmpsnooping routeaddress** command

```
IGMP Snooping Router Address
----------------------------------------
IGMP Snooping Router Mode ..... IP

Router Address List
----------------------------------------
224.0.0.4
224.0.0.6
224.0.0.80
224.0.0.43
224.0.0.23
224.0.0.15
224.0.0.60
----------------------------------------
```

Table 17-10: Parameters in the output of the **show igmpsnooping routeaddress** command

Parameter	Meaning
IGMP Snooping Router Mode	The current IGMP Snooping router mode: ALL DEFAULT MULTICASTROUTER NONE IP
Router Address List	The list of configured reserved IP multicast addresses that are considered as multicast router addresses.

Examples To show the current list of configured router multicast addresses, use the command:

```
sh igmpsn routera
```

Related Commands

add igmpsnooping routeraddress
delete igmpsnooping routeraddress
set igmpsnooping routermode

show ip igmp

Syntax `SHow IP IGMP [COUnter] [INTerface=`*`interface`*`]`

where *interface* is an interface name formed by concatenating a Layer 2 interface type, an interface instance, and optionally a hyphen followed by a logical interface number from 0 to 15. If a logical interface is not specified, 0 is assumed.

Description This command displays information about IGMP (see Figure 17-12 on page 17-74, Table 17-11 on page 17-74).

If **counter** is specified, IGMP counters is displayed (see Figure 17-13 on page 17-75, Table 17-12 on page 17-75).

If an **interface** is specified, only information about that interface is displayed. Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

Figure 17-12: Example output from the **show ip igmp** command.

```
IGMP Protocol
-------------------------------------------------------------------------------
Status ........................... Enabled
Default Query Interval ........... 125 secs
Default Timeout Interval ......... 270 secs
Disabled All-groups ports ........ 1,5,7

Last Member Query Interval ....... 10 (1/10secs)
Last Member Query Count .......... 2
Robustness Variable .............. 2
Query Response Interval .......... 100 (1/10secs)

Interface Name .......... eth0              (DR)
Other Querier timeout ... 164 secs
IGMP Proxy .............. Upstream
Group List ..............

  Group. 224.0.1.22         Last Adv. 10.194.254.254     Refresh time 184 secs
  Static Ports 3

  All Groups                Last Adv. 10.116.2.1         Refresh time 254 secs
```

Table 17-11: Parameters in the output of the **show ip igmp** command .

Parameter	Meaning
Status	Whether IGMP is enabled or disabled.
Default Query Interval	The default interval at which Host Membership Queries are sent.
Default Timeout Interval	The default interval after which entries are removed from the group database when no Host Membership Report is received.
Disabled All-groups ports	A list of ports that are disallowed from acting as an all-groups port.
Last Member Query Interval	Max Response Time inserted into Group-Specific Queries sent in response to Leave Group messages, and is also the amount of time between Group-Specific Query messages.
Last Member Query Count	The number of Group-Specific Queries sent before the router assumes there are no local members.
Robustness Variable	IGMP is robust to (Robustness Variable-1) packet losses.
Query Response Interval	The Max Response Time (in 1/10 secs) inserted into the periodic General Queries.
Interface Name	The name of an interface configured for IGMP.
Other Querier timeout	The time that remains before a multicast router decides that there is no longer another multicast router that should be the querier.
IGMP Proxy	The status of IGMP Proxy; one of “Off”, “Upstream” or “Downstream”.
Group List	A list of multicast group memberships for this interface.
Group	The group multicast address.
Last Adv.	The last host to advertise the membership report.

Table 17-11: Parameters in the output of the **show ip igmp** command (continued).

Parameter	Meaning
Refresh time	The time in seconds until the membership group is deleted when it does not receive another membership report before then.
Static Ports	A list of the static ports; a subset of the ports listed in the Ports field. The Static Ports field is only displayed for static groups on a VLAN.

Figure 17-13: Example output from the **show ip igmp counter** command.

```
IGMP Counters
-------------------------------------------------------------
Interface Name: eth0

inQuery ..... 1           outQuery ..... 5
inV1Report ..... 4
inV2Report ..... 7
inLeave ..... 0
inTotal ..... 12          outTotal ..... 5

badQuery ..... 0
badV1Report ..... 0
badV2Report ..... 0
badLeave ..... 0
badTotal ..... 1
-------------------------------------------------------------
```

Table 17-12: Parameters in the output of the **show ip igmp counter** command.

Parameter	Meaning
inQuery	The number of IGMP membership query messages that were received by the interface.
inV1Report	The number of IGMP Version 1 membership report messages that were received by the interface.
inV2Report	The number of IGMP Version 2 membership report messages that were received by the interface.
inLeave	The number of IGMP Version 2 Leave Group messages that were received by the interface.
inTotal	The total number of IGMP messages that were received by the interface.
outQuery	The number of IGMP membership query messages that were initiated by the router for the interface.
outTotal	The total number of IGMP messages that were transmitted by the router over the interface.
badQuery	The number of IGMP membership query messages with errors that were received by the interface.
badV1Report	The number of IGMP Version 1 membership report messages with errors that were received by the interface.
badV2Report	The number of IGMP Version 2 membership report messages with errors that were received by the interface.

Table 17-12: Parameters in the output of the **show ip igmp counter** command.

Parameter	Meaning
badLeave	The number of IGMP Version 2 Leave Group messages with errors that were received by the interface.
badTotal	The total number of IGMP messages with errors that were received by the interface.

Examples To display information about IGMP, use the command:

```
sh ip igmp
```

Related Commands
add ip interface
disable ip igmp
disable ip igmp interface
enable ip igmp
enable ip igmp interface
set ip igmp
set ip interface
show ip route multicast

show ip igmp debug

Syntax `SHow IP IGMP DEBug`

Description This command shows the IGMP debugging options that have been set.

Figure 17-14: Example output from **show ip igmp debug** command.

```
IGMP Debugging Information
--------------------------------------------------------------
IGMP Debugging                    Enabled
Filter by group destination       224.1.2.3
Filter by source IP               10.10.1.123
--------------------------------------------------------------
```

Table 17-13: Parameters in the output of the **show ip igmp debug** command.

Parameter	Meaning
IGMP Debugging	Whether IGMP debugging is enabled or disabled.
Filter by group destination	The Group Destination Address specified by the **destination** parameter in the **enable ip igmp debug** command. When no parameter is given, "No" is displayed instead of the IP address.
Filter by source IP	The source IP address specified by the **sourceipaddress** parameter in the **enable ip igmp debug** command. When no parameter is given, "No" is displayed instead of the IP address.

Examples To display IGMP debugging information, use the command:

```
sh ip igmp deb
```

Related Commands
disable ip igmp debug
enable ip igmp debug

show pim

Syntax

```
SHow PIM
```

Description This command displays detailed information about the PIM routing status on the router, and is equivalent to specifying all of the following commands in the following order:

```
show PIM interface
show PIM route
show PIM neighbour
show PIM counters
show PIM debug
show PIM rpcandidate
show PIM bsrcandidate
show PIM rpset
show PIM timer
show PIM config
```

Examples To display detailed PIM routing status information, use the command:

```
sh pim
```

Related Commands
disable pim
enable pim
set pim
show ip
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

show pim bsrcandidate

Syntax `SHow PIM BSRCandidate`

Description This command displays information about the router as a BSR candidate for PIM-SM (see Figure 17-15 on page 17-78, Figure 17-16 on page 17-78, Table 17-14 on page 17-78).

Figure 17-15: Example output from the **show pim bsrcandidate** command for an elected BSR.

```
PIM BSR Candidate
-------------------------------------------------------------
Preference ......................... 1
 BSR State ......................... Elected BSR
   Elected BSR IP address .......... 101.202.101.202
   Elected BSR preference .......... 1
```

Figure 17-16: Example output from the **show pim bsrcandidate** command for an unelected BSR candidate.

```
PIM BSR Candidate
-------------------------------------------------------------
BSR State ......................... Accepts Preferred BSM
  Elected BSR IP address .......... 101.202.101.202
  Elected BSR preference .......... 1
```

Table 17-14: Parameters in the output of the **show pim bsrcandidate** command.

Parameter	Meaning
Preference	The preference value for the router to be a candidate bootstrap router. The higher the number, the higher the priority. This parameter is present when the router is the elected BSR.
BSR State	The current status of the BSR; one of "Accepts Preferred BSM" (the router is available to become the BSR), or "Elected BSR" (the router is the BSR).
Elected BSR IP address	The IP address of the BSR. If the router is the BSR, this address is one of the router's addresses.
Elected BSR preference	The preference of the BSR. When the router is the BSR, this is its preference.

Examples To display information about the router as a BSR candidate, use the command:

```
sh pim bsrc
```

Related Commands

add pim bsrcandidate
delete pim bsrcandidate
disable pim
enable pim
set pim
set pim bsrcandidate
show ip

show pim
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

show pim config

Syntax `SHow PIM CONFig`

Description This command lists the command line interface commands that make up the PIM configuration (see Figure 17-17 on page 17-79).

Figure 17-17: Example output from the **show pim config** command.

```
#PIM4 configuration
#
add pim interface=eth0
add pim interface=eth1 drpriority=100
enable pim
```

Examples To display the PIM configuration, use the command:

```
sh pim conf
```

Related Commands
disable pim
enable pim
set pim
show ip
show config

show pim counters

Syntax `SHow PIM COUnters`

Description This command displays information about PIM counters (see Figure 17-18 on page 17-80, Figure 17-19 on page 17-80, Table 17-15 on page 17-81).

Figure 17-18: Example output from the **show pim counters** command for PIM Sparse Mode.

```
PIM4 Counters
---------------------------------------------------------------
Sparse Mode
---------------------------------------------------------------
eth0:
  inHello ..............  14      outHello ..............  15
  inRegister ............  0       outRegister ...........  0
  inRegisterStop ........  0       outRegisterStop .......  0
  inJP ..................  0       outJP .................  0
  inAssert ..............  0       outAssert .............  0
  inBSM .................  8       outBSM ................  3
  inCRPAdv ..............  0       outCRPAdv .............  0
  inTotal ...............  22      outTotal ..............  18

eth0 Bad:
  badHello ..............  0
  badRegister ...........  0
  badRegisterStop .......  0
  badJP .................  0
  badAssert .............  0
  badBSM ................  0
  badCRPAdv .............  0
  badTotal ..............  0

---------------------------------------------------------------
```

Figure 17-19: Example output from the **show pim counters** command for PIM Dense Mode

```
PIM4 Counters
---------------------------------------------------------------
Dense Mode
---------------------------------------------------------------
eth0:

  inHello ..............  25      outHello ..............  26
  inGraft ..............  0       outGraft ..............  0
  inGraftAck ............  0       outGraftAck ...........  0
  inJP ..................  0       outJP .................  0
  inAssert ..............  0       outAssert .............  0
  inSRM .................  0       outSRM ................  0
  inTotal ...............  25      outTotal ..............  26

eth0 Bad:
    badHello ............  0
    badGraft ............  0
    badGraftAck .........  0
    badJP ...............  0
    badAssert ...........  0
    badTotal ............  0
---------------------------------------------------------------
```

Table 17-15: Parameters in the output of the **show pim counters** command.

Parameter	Meaning
inHello	The number of PIM Hello messages received by the interface.
inRegister	The number of PIM Register messages that were received by the interface. This parameter is displayed for PIM-SM interfaces only.
inRegisterStop	The number of PIM Register Stop messages received by the interface. This parameter is displayed for PIM-SM interfaces only.
inGraft	The number of PIM Graft messages received by the interface. This parameter is displayed for PIM-DM interfaces only.
inGrackAck	The number of PIM Graft acknowledgement messages that were received by the interface. This parameter is displayed for PIM-DM interfaces only.
inJP	The number of PIM Join and Prune messages received by the interface.
inAssert	The number of PIM Assert messages received by the interface.
inBSM	The number of PIM BootStrap messages received by the interface. This parameter is displayed for PIM-SM interfaces only.
inCRPAdv	The number of PIM Candidate RP Advertisement messages received by the interface. This parameter is displayed for PIM-SM interfaces only.
inSRM	The number of PIM State Refresh messages received by the interface. This parameter is displayed for PIM-DM interfaces only.
inTotal	The total number of PIM messages received by the interface.
outHello	The number of PIM Hello messages transmitted by the interface.
outRegister	The number of PIM Register messages transmitted by the interface. This parameter is displayed for PIM-SM interfaces only.
outRegisterStop	The number of PIM Register Stop messages transmitted by the interface. This parameter is displayed for PIM-SM interfaces only.
outGraft	The number of PIM Graft messages transmitted by the interface. This parameter is displayed for PIM-DM interfaces only.
outGrackAck	The number of PIM Graft acknowledgement messages transmitted by the interface. This parameter is displayed for PIM-DM interfaces only.
outJP	The number of PIM Join and Prune messages transmitted by the interface.
outAssert	The number of PIM Assert messages transmitted by the interface.

Table 17-15: Parameters in the output of the **show pim counters** command.

Parameter	Meaning
outBSM	The number of PIM BootStrap messages transmitted by the interface. This parameter is displayed for PIM-SM interfaces only.
outCRPAdv	The number of PIM Candidate RP Advertisement messages transmitted by the interface. This parameter is displayed for PIM-SM interfaces only.
outSRM	The number of PIM State Refresh messages transmitted by the interface. This parameter is displayed for PIM-DM interfaces only.
outTotal	The total number of PIM messages that were transmitted by the interface.
badHello	The number of PIM Hello messages with errors that were received by the interface.
badRegister	The number of PIM Register messages with errors received by the interface. This parameter is displayed for PIM-SM interfaces only.
badRegisterStop	The number of PIM Register Stop messages with errors received by the interface. This parameter is displayed for PIM-SM interfaces only.
badGraft	The number of PIM Graft messages with errors received by the interface. This parameter is displayed for PIM-DM interfaces only.
badGrackAck	The number of PIM Graft acknowledgement messages with errors received by the interface. This parameter is displayed for PIM-DM interfaces only.
badJP	The number of PIM Join and Prune messages with errors received by the router.
badAssert	The number of PIM Assert messages with errors received by the interface.
badBSM	The number of PIM BootStrap messages with errors received by the interface. This parameter is displayed for PIM-SM interfaces only.
badCRPAdv	The number of PIM Candidate RP Advertisement messages with errors received by the interface. This parameter is displayed for PIM-SM interfaces only.
badSRM	The number of PIM State Refresh messages with errors received by the interface. This parameter is displayed for PIM-DM interfaces only.
badTotal	The total number of PIM messages with errors received by the interface.

Examples To display information about PIM counters, use the command:

```
sh pim cou
```

Related Commands

disable pim
enable pim
set pim
show ip
show pim

show pim bsrcandidate
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

show pim debug

Syntax

```
SHow PIM DEBug
```

Description

This command displays the list of PIM interface debugging options (see Figure 17-20 on page 17-83, Table 17-16 on page 17-83).

Figure 17-20: Example output from the **show pim debug** command.

```
PIM Debug Options
------------------------------------------------------------
Debug Options Enabled: Join, Assert

Logging Options Enabled : status

Trapping Options Enabled: none

Info (1097049): The PIM module is not enabled.
```

Table 17-16: Parameters in the output of the **show pim debug** command.

Parameter	Meaning
Debug Option	A comma-separated list of the PIM debugging options that are enabled, or "None" if debugging is disabled, or "All" if all debugging is enabled. Options are listed in Table 17-2 on page 17-55.

Examples

To display a list of enabled PIM interface debugging options, use the command:

```
sh pim deb
```

Related Commands

disable pim
disable pim debug
enable pim
enable pim debug
set pim
show ip
show pim
show pim bsrcandidate
show pim counters
show pim interface
show pim neighbour

show pim route
show pim rpcandidate
show pim rpset
show pim timer

show pim interface

Syntax `SHow PIM INTerface`

Description This command displays information about all PIM interfaces and their designated router status (see Figure 17-21 on page 17-84, Figure 17-22 on page 17-84, Table 17-17 on page 17-85). Valid interfaces are:

- eth (e.g. eth0, eth0-1)
- PPP (e.g. ppp0, ppp1-1)
- VLAN (e.g. vlan1, vlan1-1)
- FR (e.g. fr0, fr0-1)

Figure 17-21: Example output from the **show pim interface** command for PIM Sparse Mode.

```
PIM4 Sparse mode Interface Table
------------------------------------------------------------
Interface ........................ eth0
  IP address ..................... 172.128.71.25
    DR election by ............... DR priority
    DR priority .................. 10
    DR winner .................... Me
  Hello interval ................. 30

Interface ........................ eth1
  IP address ..................... 172.128.72.33
    DR election by ............... DR priority
    DR priority .................. 1
    DR winner .................... 172.128.72.14
  Hello interval ................. 30
```

Figure 17-22: Example output from the **show pim interface** command for PIM Dense Mode.

```
PIM4 Dense mode Interface Table
------------------------------------------------------------
Interface ........................ eth1
  IP address ..................... 192.168.0.111
  State refresh capable .......... No
  Hello interval ................. 30
```

Table 17-17: Parameters in the output of the **show pim interface** command.

Parameter	Meaning
Interface	IP interfaces running PIM processes.
Dlci	The data link circuit identifier for a Frame Relay interface.
IP Address	The IP address on this interface.
DR election by	How this interface elects a DR; one of "DR priority "(the DR priority is transmitted in Hello messages and election is by priority), or "IP address" (the DR priority is not transmitted in Hello messages so election is by IP address).
DR priority	The priority for the DR candidate to become the PIM designated router. A candidate with a higher priority is more likely to become the DR.
DR Winner	The IP address of the PIM designated router for the interface, or "me" when this router is the designated router.
State refresh capable	Whether this interface originates and processes State Refresh messages for PIM-DM.
Hello interval	The interval, in seconds, at which the router sends PIM Hello messages on this interface. The value 65535 indicates that the Hello message never expires.

Examples To display information about all PIM interfaces, use the command:

```
sh pim int
```

Related Commands

add pim interface
delete pim interface
disable pim
enable pim
reset pim interface
set pim interface
set pim
show ip in Chapter 14, Internet Protocol (IP)
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset
show pim timer

show pim neighbour

Syntax `SHow pim NEIghbour`

Description This command displays information about the PIM Neighbour Table (see Figure 17-23 on page 17-86, Figure 17-24 on page 17-86, Table 17-18 on page 17-86).

Figure 17-23: Example output from the **show pim neighbour** command for PIM Sparse Mode.

```
PIM4 Sparse mode Neighbour Table
------------------------------------------------------------
Interface ........................ eth0
  IP Address ..................... 137.39.3.93
    DR Priority .................. 1
    Neighbour Liveness Timer ..... 82
```

Figure 17-24: Example output from the **show pim neighbour** command for PIM Dense Mode.

```
PIM4 Dense mode Neighbour Table
------------------------------------------------------------
Interface ........................ eth0
  IP Address ..................... 192.168.57.2
    Neighbour Liveness Timer ..... 105
    Is state refresh capable ..... No
```

Table 17-18: Parameters in the output of the **show pim neighbour** command.

Parameter	Meaning
Interface	The interface to which the PIM neighbour is connected.
IP Address	The IP address of the neighbour.
DR Priority	The priority for this neighbour to become the designated router for the subnetwork.
Neighbour Liveness Timer	The time in seconds until the neighbour is removed from the neighbour table.
Is state refresh capable	Whether the neighbour originates and processes State Refresh messages for PIM-DM.

Examples To display information about the PIM Neighbour Table, use the command:

```
sh pim nei
```

Related Commands

disable pim
enable pim
set pim
show ip
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface

show pim route

Syntax `SHow PIM ROUte`

Description This command displays information about the internal PIM routing table, for PIM Sparse Mode (see Figure 17-25 on page 17-87, Figure 17-26 on page 17-88, Table 17-19 on page 17-88) and/or Dense Mode (Figure 17-27 on page 17-91, Table 17-20 on page 17-91).

Figure 17-25: Example output from the **show pim route** command for PIM Sparse Mode, when the router is the RP.

```
PIM4 Sparse Mode Tree Information Base
--------------------------------------------------------------
Group ............................ 224.1.1.1
  Type ............................ (*,G)
    RP Address .................... I am the RP
    Expiry time ................... 630
    Join/prune time ............... 0
    Immediate output interfaces .. eth2

  Type ............................ (S,G)
    Source ........................ 192.168.0.1
    RPF Neighbour to Src ......... 192.168.1.1
    RPF Interface to Src ......... eth0
    Expiry time ................... 180
    Keepalive time ................ 160
    Join/prune time ............... 0
    Register time ................. 0
    SPT bit ....................... Unset
    Inherited output interfaces .. eth2
    Immediate output interfaces .. None

  Type ............................ (S,G,rpt)
    Source ........................ 192.168.0.1
    RP Address .................... I am the RP
    Expiry time ................... 180
    Override time ................. 0
    Inherited output interfaces .. eth2

  Type ............................ (*,*,RP)
    RP Address .................... I am the RP
    Expiry time ................... 210
    Join/prune time ............... 0
    Immediate output interfaces .. None
```

Figure 17-26: Example output from the **show pim route** command for PIM Sparse Mode, when the router is not the RP.

```
PIM4 Sparse Mode Tree Information Base
--------------------------------------------------------------
Group ............................ 224.1.1.1
  Type ............................ (*,G)
    RP Address .................... 192.168.1.1
    RPF Neighbour to RP ........... 192.168.2.1
    RPF Interface to RP ........... eth0
    Expiry time ................... 630
    Join/prune time ............... 0
    Immediate output interfaces .. eth2

  Type ............................ (S,G)
    Source ........................ 192.168.0.1
    RPF Neighbour to Src ......... Directly connected
    RPF Interface to Src ......... eth1
    Expiry time ................... 230
    Keepalive time ................ 210
    Join/prune time ............... 0
    Register time ................. 21
    SPT bit ....................... Unset
    Inherited output interfaces .. eth2
    Immediate output interfaces .. eth2

  Type ............................ (S,G,rpt)
    Source ........................ 192.168.0.1
    RP Address .................... 192.168.1.1
    Expiry time ................... 230
    Override time ................. 0
    Inherited output interfaces .. eth2

 Type ............................ (*,*,RP)
    RP Address .................... 192.168.1.1
    Next hop to RP ................ 192.168.2.1
    RPF Interface to RP ........... eth0
    Expiry time ................... 210
    Join/prune time ............... 0
    Immediate output interfaces .. None
```

Table 17-19: Parameters in the output of the **show pim route** command for PIM Sparse Mode.

Parameter	Entry Type	Parameter for Entry Type	Meaning
Group			The IP address of the multicast group.
Type			The type of entry in the Tree Information Base.
	(*,G)		The entry for traffic from any source to a particular group.
		RP Address	The IP address of the Rendezvous Point for the group.
		RPF Neighbour to RP	The address of the PIM neighbour to the RP, taking into account any PIM assert messages. Packets from the RP would be received from this neighbour.
		RPF Interface to RP	The interface on which packets from the RP would be received.

Table 17-19: Parameters in the output of the **show pim route** command for PIM Sparse Mode. (continued)

Parameter	Entry Type	Parameter for Entry Type	Meaning
		Expiry time	The time remaining until this entry is deleted, in seconds. A zero value indicates that the timer is not running. This timer decrements when there are no (S,G) entries.
		Join/prune time	The Join/prune timer in seconds. When the router sees a Prune message on the correct upstream interface, and it still needs to receive traffic via that rp tree, it sends a Join message when this timer expires. A zero value indicates that the timer is not running.
		Immediate output interfaces	The interfaces with downstream routers or IGMP hosts that are interested in this (*,G) entry.
	(S,G)		The entry for traffic from a particular source to a particular group.
		Source	The IP address of the multicast sender.
		RPF Neighbour to Src	The address of the PIM neighbour to the source, taking into account any PIM assert messages. Packets from the source would be received from this neighbour. "Directly connected" indicates that the source is directly connected to the router.
		RPF Interface to Src	The interface on which packets from the source would be received, if the source is in this multicasting domain.
		Expiry time	The time remaining until this entry is deleted, in seconds. A zero value indicates that the timer is not running. The expiry time is 20 seconds longer than the Keepalive time.
		Keepalive time	The Keepalive timer in seconds. A zero value indicates that the timer is not running because no data is being received. The timer is reset when data is received.
		Join/prune time	The Join/prune timer, in seconds. When the router sees a Prune message on the correct upstream interface, and it still needs to receive traffic via that sp tree, it sends a Join message when this timer expires. A zero value indicates that the timer is not running.
		Register time	The Register suppression time, in seconds. When this timer reaches the Register Probe Time, a Null Register message is sent to the RP.
		SPT bit	Whether forwarding is set on the Shortest Path Tree.
		Inherited output interfaces	The interfaces to forward (S,G) data to.
		Immediate output interfaces	The interfaces with downstream routers or IGMP hosts that are interested in this (S,G) data.

Table 17-19: Parameters in the output of the **show pim route** command for PIM Sparse Mode. (continued)

Parameter	Entry Type	Parameter for Entry Type	Meaning
	(S,G, rpt)		The entry that is used for suppressing traffic on the RP tree from a particular source to a particular group. This entry applies when the traffic is known to be flowing down the shortest path tree, so the traffic is no longer needed via the RP tree.
		Source	The IP address of the multicast sender.
		RP Address	The IP address of the Rendezvous Point for the group.
		Expiry time	The time remaining until this entry is deleted, in seconds.The expiry time is 20 seconds longer than the (S,G) Keepalive time.
		Override time	The Override timer in seconds. When the router sees a Prune message on the correct upstream interface, and it still needs to receive traffic via that rp tree, it sends a Join message when this timer expires. A zero value indicates that the timer is not running.
		Inherited output interfaces	The interfaces that still require (S,G) data via the RP tree.
	(*,*, RP)		The entry for handling multicast traffic to and from a network that is running a different multicast protocol. This entry applies when the router is a PIM multicast border router (PMBR).
		RP Address	The IP address of the Rendezvous Point for the group.
		Next hop to RP	The address of the next routing device on the best unicast routing path to the RP.
		RPF Interface to RP	The interface on which packets from the RP would be received.
		Expiry time	The time remaining until this entry is deleted, in seconds.
		Join/Prune time	The Join/prune timer in seconds. When the router sees a Prune message on the correct upstream interface, and it still needs to receive traffic via that rp tree, it sends a Join message when this timer expires. A zero value indicates that the timer is not running.
		Immediate output interfaces	The interfaces with downstream routers that are interested in this (*,*,RP) entry.

Figure 17-27: Example output from the **show pim route** command for PIM Dense Mode.

```
PIM4 Dense Mode Tree Information Base
-------------------------------------------------------------
  Source ....................... 172.95.1.1
  Group ........................ 238.1.2.3
    RPF Neighbour to Src ....... Directly connected
    RPF Interface to Src ....... eth0
    Source Alive time ..... 200
    Expiry time ..... 220
    Prune override time ..... 0
    Prune limit time ..... 0
    Immediate output interfaces .. eth1

  Source ....................... 172.96.2.1
  Group ........................ 238.1.2.3
    RPF Neighbour to Src ....... 192.168.57.1
    RPF Interface to Src ....... eth1
    Keep Alive time ..... 200
    Expiry time ..... 220
    Prune override time ..... 0
    Prune limit time ..... 50
    Immediate output interfaces .. eth1
```

Table 17-20: Parameters in the output of the **show pim route** command for PIM Dense Mode.

Parameter	Meaning
Source	The IP address of the multicast sender.
Group	The IP address of the multicast group.
RPF Neighbour to Src	The address of the PIM neighbour to the source, taking into account any PIM assert messages. Packets from the source would be received from this neighbour. "Directly connected" indicates that the source is directly connected to the router.
RPF Interface to Src	The interface on which the router expects to receive traffic from the source.
Keep Alive time	The Keepalive timer in seconds. A zero value indicates that the timer is not running because no data is being received. The timer is reset when data is received.
Source Alive time	An alive timer in seconds that is the equivalent of the Keepalive timer but applies to directly connected sources. A zero value indicates that the timer is not running because no data is being received. The timer is reset when data is received.
Expiry time	The time remaining until this entry is deleted, in seconds. The expiry time is 20 seconds longer than the (S,G) Keepalive or Sourcealive time.
Prune override time	The Prune override timer, in seconds. When the router sees a Prune message on the correct upstream interface, and it still needs to receive traffic, it sends a Join message when this timer expires. A zero value indicates that the timer is not running.
Prune limit time	The Prune limit, in seconds. A zero value indicates that the timer is not running. The router cannot send a data-triggered prune until this timer expires.
Immediate output interfaces	The interfaces with routers or IGMP hosts that are interested in this (S,G) data.

Examples To display information about the internal PIM routing table, use the command:

```
sh pim rou
```

Related Commands

disable pim
enable pim
set pim
show ip
show ip route multicast
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim rpcandidate
show pim rpset
show pim timer

show pim rpcandidate

Syntax `SHow PIM RPCandidate`

Description This command displays information about multicast groups for which the router is a PIM-SM Rendezvous Point candidate (see Figure 17-28 on page 17-92, Table 17-21 on page 17-92).

Figure 17-28: Example output from the **show pim rpcandidate** command.

```
PIM4 RP Candidate
-------------------------------------------------------------
Priority ....................... 192
Interface .....................vlan1
    Group address/Mask ........... 224.1.1.1 / 255.255.255.255
     Group address/Mask ........... 224.2.2.0 / 255.255.255.0
```

Table 17-21: Parameters in the output of the **show pim rpcandidate** command.

Parameter	Meaning
Priority	The priority for the router to become the Rendezvous Point for any multicast groups.
Group Address	The multicast groups associated with the specified Rendezvous Point.
Mask	The mask for the address.
Interface	The interface the switch will advertise itself as when advertising as a Rendezvous Point for multicast groups.

Examples To display information about multicast groups for which the router is a Rendezvous Point candidate, use the command:

```
sh pim rpc
```

Related Commands

add pim rpcandidate
delete pim rpcandidate
disable pim
enable pim
set pim
set pim rpcandidate
show ip
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpset
show pim timer

show pim rpset

Syntax

```
SHow PIM RPSet
```

Description

This command displays the static group-to-RP mapping (see Figure 17-29 on page 17-93, Table 17-22 on page 17-93), followed by the elected bootstrap router's current set of RP candidates and the groups they are configured for (see Figure 17-30 on page 17-94, Table 17-23 on page 17-94). It applies to PIM-SM only.

Figure 17-29: Example output from the **show pim rpset** command when the RP is statically configured.

```
PIM4 Static RP Mapping
-------------------------------------------------------------
RP Address ......................... 192.168.2.1
    Group address/Mask ............... 239.1.0.0 /
255.255.0.0
```

Table 17-22: Parameters in the output of the **show pim rpset** command when the RP is statically configured.

Parameter	Meaning
RP address	The IP address of the router that is statically configured as the RP for the following group(s).
Group address	The IP address of the multicast group.
Mask	The mask for the multicast group address.

Figure 17-30: Example output from the **show pim rpset** command when the RP is determined using the bootstrap mechanism.

```
PIM4 RP Set Information
-------------------------------------------------------

Group address/Mask ........ 224.1.1.1 / 255.255.255.255
    RP Candidate address .. 192.168.1.1
      Priority ............ 192
    RP Candidate address .. 192.168.2.1
      Priority ............ 180
```

Table 17-23: Parameters in the output of the **show pim rpset** command when the RP is determined using the bootstrap mechanism.

Parameter	Meaning
Group address	The IP address of the multicast group.
Mask	The mask for the multicast group address.
RP Candidate address	The IP addresses of each RP Candidate for the multicast group and mask pair.
Priority	The priority for the RP candidate to become the RP. A candidate with a lower priority is more likely to become the RP.

Examples

To display information about multicast group and mask pairs, use the command:

```
sh pim rps
```

Related Commands

disable pim
enable pim
set pim
show ip
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim timer

show pim staterefresh

Syntax `SHow PIM STATerefresh`

Description This command displays the internal State Refresh table for PIM-DM (see Figure 17-31 on page 17-95, Table 17-24 on page 17-95).

Figure 17-31: Example output from the **show pim staterefresh** command.

```
PIM4 Dense Mode State Refresh
-------------------------------------------------------------
  Source ............................. 172.95.2.1
  Group .............................. 238.1.2.3
    Originator state ................. Orginator
      Direct Connect to source on ..... eth1
      Source alive timer .............. 200
      State refresh timer ............. 50


  Source ............................. 172.96.2.1
  Group .............................. 238.1.2.3
    Originator state ................. Not Originator
```

Table 17-24: Parameters in the output of the **show pim staterefresh** command.

Parameter	Meaning
Source	The IP address of the multicast sender.
Group	The IP address of the multicast group.
Originator state	Whether the router can act as a state refresh message originator. A router acts as an originator when the source is directly connected.
Direct Connect to source on	The interface the source is connected to.
Source alive timer	An alive timer in seconds for directly connected sources. A zero value indicates that the timer is not running because no data is being received. The timer is reset when data is received.
State refresh timer	The time before the next state refresh message is sent, in seconds.

Examples To display the internal State Refresh table, use the command:

```
sh pim stat
```

Related Commands

disable pim
enable pim
set pim
show ip
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset

show pim timer

Syntax `SHow PIM TIMer`

Description This command displays information about timer intervals for PIM operations (see Figure 17-32 on page 17-96, Table 17-25 on page 17-96).

Figure 17-32: Example output from the **show pim timer** command.

```
PIM Timers
-------------------------------------------------------------
Join/Prune interval ................... 60
Register probe time ................... 5
Register suppression time ............. 60
Keep Alive time ....................... 210
BSM interval .......................... 60
RP adv interval ....................... 60
Prune hold time ....................... 60
Source Alive time ..................... 210
State refresh interval ................ 60
```

Table 17-25: Parameters in the output of the **show pim timer** command.

Parameter	Meaning
Join/Prune Interval	The time interval in seconds at which the router sends Join/Prune messages.
Register Probe time	The time interval in seconds that the DR waits for another Register Stop message after sending a Null Register message to the RP.
Register Suppression time	The time interval in seconds at which the sender's DR sends Null Register messages to the group's RP.
Keep Alive time	Length in seconds that the Join state for a particular Source and Group pair is maintained in the absence of data for that pair.
BSM interval	Length in seconds that the router sends bootstrap messages when it is the bootstrap router in the domain.
RP adv interval	Length in seconds that the router sends C-RP-Advertisements.
Prune hold time	Length in seconds that upstream routers maintain the Prune state.
Source Alive time	Length in seconds that a router acting as a State Refresh Originator is active in the absence of data packets from the source.
State refresh interval	Length in seconds that a router sends State Refresh messages.

Examples To display information about timer intervals for PIM operations, use the command:

```
sh pim tim
```

Related Commands

disable pim
enable pim
set pim
show ip
show pim
show pim bsrcandidate
show pim counters
show pim debug
show pim interface
show pim neighbour
show pim route
show pim rpcandidate
show pim rpset