AlliedWare™ OS

How To | Configure Port–IP Binding

Introduction

This How To Note describes how to configure switches to operate in a way that is frequently required by service providers. These requirements may be as follows:

- The switch is to act as a Layer 2 switch
- The service provider's clients are attached to the 10/100 ports of the switch, and they require one client on each 10/100 port.
- The switch's gigabit ports provide the connection back into the service provider's network.
- The clients have all been statically allocated IP addresses by the service provider.
- The service provider wants to ensure that:
  - Any given client cannot use any IP address other than the address that has been allocated to them.
  - No packets destined for any given client can be seen by any other client.

In summary, the service provider's requirement is that only one IP address can be used on any port. This can be referred to as **port–IP binding**.

Contents

Which product and software version(s) does this Note apply to?

The information provided here applies to:

- Products: Rapier, Rapier i, AT-8700XL, and AT-8600 series switches
- Software versions: 2.6.1 and above

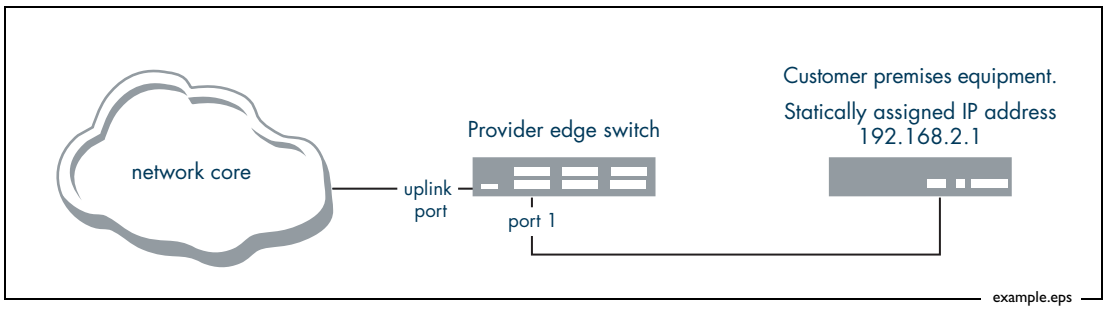
Port–IP binding with software versions 2.7.6 and later

With software versions 2.7.6 and later, DHCP snooping is a better way to achieve port–IP binding than the solution in this How To Note.

You can use DHCP snooping even if your network uses static IP addresses, by creating static DHCP entries.

For more information about DHCP snooping, see *How To Use DHCP Snooping, Option 82, and Filtering on AT-8800, AT-8600, AT-8700XL and Rapier Switches*. This Note is available from the How To Library at www.alliedtelesis.com/resources/literature/howto.aspx.

Port–IP binding scenario



The switch needs to be configured so that:

- Port 1 will not accept any IP packets that do not have source IP address 192.168.2.1
- Port 1 will not transmit any IP packets that do not have a destination address of 192.168.2.1
- The switch will not send IP packets with destination IP address 192.168.2.1 to any port except port 1.

Basic port–IP binding configuration

The first part of the service provider's requirement—"Any given client cannot use any IP address other than the address that has been allocated to them"—is quite simple to achieve by using a set of hardware filters as shown below:

```
create classifier=1 iport=1 ipsa=192.168.2.1/32
create classifier=31 eport=1 ipda=192.168.2.1/32

create classifier=2 iport=2 ipsa=192.168.2.2/32
create classifier=32 eport=2 ipda=192.168.2.2/32
...
create classifier=24 iport=24 ipsa=192.168.2.24/32
create classifier=54 eport=24 ipda=192.168.2.24/32

create classifier=806 prot=0806

add switch hwf classifier=1 action=nodrop nomatchaction=discard
add switch hwf classifier=2 action=nodrop nomatchaction=discard
...
add switch hwf classifier=24 action=nodrop nomatchaction=discard

add switch hwf classifier=31 action=nodrop nomatchaction=discard
add switch hwf classifier=32 action=nodrop nomatchaction=discard
...
add switch hwf classifier=54 action=nodrop nomatchaction=discard

add switch hwf classifier=806 action=nodrop
```

These filters have the effect that only one particular IP address is allowed to pass into or out of each of ports 1-24. The final filter allows ARP to always be sent on all ports.

The tricky remaining step in the configuration

Unfortunately, the configuration above does not completely cover the second part of the requirement—"No packets destined for any given client can ever be seen by any other client".

When a port or device goes down

The failure to cover this requirement occurs in the very specific circumstance that a particular client has disconnected their device from the switch, or their device has gone down. When the client device no longer has an active connection to the switch, then the port via which the client connected to the switch goes into a down state. One of the things that the switch does when a port goes into a down state is to remove from the Forwarding Data Base any MAC addresses that had been learnt on that port. Hence, the client's MAC address is no longer in the FDB.

So, the next time a packet arrives at the switch destined for the MAC address of the 'down' client, the switch will not have that MAC address in the FDB, and will experience what is known as a **destination lookup failure** (DLF). When a DLF occurs, the switch will forward the packet to all ports that are in the same VLAN as the ingress port. In this case, the switch is configured with only one VLAN, so the DLF packet will be flooded to all ports.

Now, it might appear initially that the hardware filters based on the **eport** classifiers like **create classifier=32 eport=2 ipda=192.168.2.2/32** would block the DLF packet from actually being broadcast to all ports. However, the operation of the switching ASIC is such that a DLF packet will not be blocked by a filter that matches on the **eport** parameter. This is because the **eport** parameter in a filter can only be considered after a packet has successfully completed the destination lookup process (and so the egress port for the packet has been uniquely identified). A DLF packet has, of course, failed the destination lookup process, so there cannot be a value of **eport** associated with the packet.

Add new filters

So, a new set of filters have to be added to the configuration to deal with this situation. The filters take the form:

```
create classifier=61 ipda=192.168.2.1/32
add switch hwf classifier=61 action=sendnonunicasttoport port=1

create classifier=62 ipda=192.168.2.2/32
add switch hwf classifier=62 action=sendnonunicasttoport port=2

...

create classifier=84 ipda=192.168.2.24/32
add switch hwf classifier=84 action=sendnonunicasttoport port=24
```

It might initially seem odd that the value of the **action** parameter is "send non-unicast-to-port". But, in fact, this action is applied to any packet that has not been assigned a specific egress port—i.e. packets that are broadcast, multicast or DLF.

The effect, then, of the hardware filter configured above is that any packet with destination IP address 192.168.2.1 that has suffered a DLF will still be forcibly sent to port 1, and not broadcast to all ports. Given that port 1 is typically DOWN when there is a DLF for packets destined to 192.168.2.1, the effect of the hardware filter will, in fact, typically be that the packet will be dropped. So, this prevents packets that are destined for one client ever being sent to other clients.

The complete filter configuration, therefore, becomes:

```
create classifier=1 iport=1 ipsa=192.168.2.1/32
create classifier=31 eport=1 ipda=192.168.2.1/32
create classifier=61 ipda=192.168.2.1/32


create classifier=2 iport=2 ipsa=192.168.2.2/32
create classifier=32 eport=2 ipda=192.168.2.2/32
create classifier=62 ipda=192.168.2.2/32

...


create classifier=24 iport=24 ipsa=192.168.2.24/32
create classifier=54 eport=24 ipda=192.168.2.24/32
create classifier=82 ipda=192.168.2.24/32

create classifier=806 prot=0806

add switch hwf classifier=1 action=nodrop nomatchaction=discard
add switch hwf classifier=2 action=nodrop nomatchaction=discard
...
add switch hwf classifier=24 action=nodrop nomatchaction=discard

add switch hwf classifier=31 action=nodrop nomatchaction=discard
add switch hwf classifier=32 action=nodrop nomatchaction=discard
...
add switch hwf classifier=54 action=nodrop nomatchaction=discard

add switch hwf classifier=61 action=sendnonunicasttoport port=1
add switch hwf classifier=62 action=sendnonunicasttoport port=2
...
add switch hwf classifier=84 action=sendnonunicasttoport port=24

add switch hwf classifier=806 action=nodrop
```

Classifier numbers

The classifiers are numbered by taking the port number and adding 0, 30 and 60. For example, classifiers for port 1 are 1, 31 and 61. Classifiers for port 15 would be 15, 45 and 75. Classifiers for port 24 are 24, 54 and 84.

A note about 48-port switches

On 48-port switches running software version 2.8.1 or above, this solution requires the switch to be in port-specific filtering (PSF) mode. PSF mode is the default. You can see which mode your switch is in by using the command:

```
show switch hwfilter
```

and checking the “Mode” field.

If your switch is currently in non-port-specific filtering mode, you can change it by using the command:

```
set switch hwfilter mode=psf
```

Configuring the filters in the correct order

You will notice above that the order of configuration of the filters has been done in a very specific way. All the filters based on classifiers using the **iport** parameter are configured first, then filters based on classifiers using the **eport** parameters, then the filters that redirect 'non-unicast' packets to a particular egress port.

The reason for this ordering relates to the structure of the tables in the ASIC that hold the filter definitions. There are 5 tables in the ASIC:

- One for filters that examine traffic arriving via ports 1-8
- One for filters that examine traffic arriving via ports 9-16
- One for filters that examine traffic arriving via ports 17-24
- One for filters that examine traffic arriving via the first gigabit port
- One for filters that examine traffic arriving via the second gigabit port

The filters that are based on classifiers that specify **iport** will only examine packets arriving via that particular port. So, those filters will only be added to one of the 5 tables (the table that operates on the block of ports to which the **iport** belongs). But, the other filters are not ingress-port specific, so they must go into all 5 tables, as the packets which match the filters could arrive via any port of the switch.

You might have decided to configure the classifier-based hardware filters in such a way that, for example, all the filters relating to port 1 were configured first, then all the filters relating to port 2, and so on, like:

```
add switch hwf classifier=1 action=nodrop nomatchaction=discard
add switch hwf classifier=31 action=nodrop nomatchaction=discard
add switch hwf classifier=61 action=sendnonunicasttoport port=1
...
add switch hwf classifier=24 action=nod nomatchaction=discard
add switch hwf classifier=54 action=nod nomatchaction=discard
add switch hwf classifier=84 action=sendnonunicasttoport port=24
```

But, with this order, the switch would add the first filter just to the table relating to ports 1-8. The next filter would be added to all tables, and so on.

The net effect would be that the start of the table for the first block of ports would have an **iport** filter, followed by an **eport** filter, followed by a **sendnonunicast** filter; whereas the start of the table for the second block of ports would just have an **eport** filter, followed by a **sendnonunicast** filter, and so on.

This situation of differing filter ordering for different blocks of ports could lead to subtle filtering problems, particularly if any filters were deleted, or new ones added.

But, if you order the filters such that all the **iport** filters are configured, then all the **eport** filters, then all the **sendnonunicast** filters, this gives all the tables a similar ordering of filters, and the problems will not be seen.

Maintaining the configuration

Deleting filters, or adding new filters, could alter the ordering of the filters in the tables, and so cause unexpected filtering decisions in the switch.

So, it is important to make sure that you maintain the filtering ordering of all the **iport** filters, followed by all the **eport** filters, followed by all the **sendnonunicast** filters.

So, if filters are to be added or deleted, the best way to do so would be to edit the configuration script and then reboot.

A somewhat less service-affecting approach would be to have the filter part of the switch configuration written in a script that is separate from the main config script. This "filter" script could have a set of commands to delete all hardware filters, followed by a set of commands to create the desired set of hardware filters. Then, if the set of filters was to be changed, this file could be edited and re-run. The script would, then, clear out all the existing filters, so that the filter tables were clean, and then write in the new set of filters. This way, the content of the filtering tables could be kept reliably consistent.

USA Headquarters | 19800 North Creek Parkway | Suite 200 | Bothell | WA 98011 | USA | T: +1 800 424 4284 | F: +1 425 481 3895
European Headquarters | Via Motta 24 | 6830 Chiasso | Switzerland | T: +41 91 69769.00 | F: +41 91 69769.11
Asia-Pacific Headquarters | 11 Tai Seng Link | Singapore | 534182 | T: +65 6383 3832 | F: +65 6383 3830
www.alliedtelesis.com

C613-16054-00 REV B