

## Release Note

# Software Release 2.4.1

## For AR300, AR400, and AR700 Series Routers, Rapier and Rapier *i* Series Switches, and AR800 Series Modular Switching Routers

Introduction .....	2
Hardware Platforms .....	2
New Features .....	3
Graphical User Interface (GUI) .....	4
COPY Command .....	5
Enhancements to Switching .....	5
Generic Packet Classifiers .....	6
Hardware Packet Filters .....	8
Protocol Type Parameter for Layer 3 Hardware Filters .....	9
Quality of Service (QoS) .....	10
Access Control Lists (ACLs) .....	11
Overlapping VLANs/STPs .....	12
Configurable MRU and MAXLINKS options .....	16
IP Route Filtering .....	17
Ping Timeout .....	19
TCP Listen Ports Closed by Default .....	19
IPv6 Routing Enhancements .....	19
IGMP Proxy .....	22
Setting IGMP Thresholds .....	24
Multicast VLAN Registration (MVR) .....	25
Enhancements to IP Multicast Switching .....	26
IPv6 Multicasting .....	26
OSPF/RIP Static Export Control Parameter .....	31
Encryption Key Support for IPv6 .....	32
AppleTalk Filtering .....	33
DHCP Extended Client ID .....	36
IPsec Support for IPv6 .....	38
3DES Outer CBC Encryption Mode .....	39
BGP-4 Triggers .....	40
Availability .....	41
Installation .....	41

## Introduction

---

Allied Telesyn announces the release of Software Release 2.4.1 for AR300 Series routers, AR400 Series routers, AR700 Series routers, Rapier and Rapier *i* Series layer 3 switches, and AR800 Series modular switching routers. This Release Note describes the new features in Software Release 2.4.1 since Software Release 2.3.2 for AR400 and AR700 Series routers, and Software Release 2.3.1 for AR300 and AR700 Series routers, Rapier and Rapier *i* Series layer 3 switches, and AR800 Series modular switching routers.

For information about Software Release 2.4.1 on AR100 Series routers, see the Release Note “*Software Release 2.4.1 for AR100 Series Internet Routers*” (document number C613-10339-00 REV A).

This release note should be read in conjunction with the Quick Start Guide, User Guide, Hardware Reference, and Software Reference for your router or switch. These documents can be found on the Documentation and Tools CD-ROM packaged with your router or switch, or on the support site at [www.alliedtelesyn.co.nz/documentation/documentation.html](http://www.alliedtelesyn.co.nz/documentation/documentation.html)



---

**WARNING:** *Information in this document is subject to change without notice and does not represent a commitment on the part of Allied Telesyn International. While every effort has been made to ensure that the information contained within this document and the features and changes described are accurate, Allied Telesyn International can not accept any type of liability for errors in, or omissions arising from the use of this information.*

---

## Hardware Platforms

---

Software Release 2.4.1 supports the following existing hardware platforms:

- AR300 Series Routers
- AR720 and AR740 Routers
- AR410 Routers
- Rapier Series Layer 3 Switches
- Rapier *i* Series Layer 3 Switches

Software Release 2.4.1 adds support for the following new hardware platforms:

- AR725 and AR745 Routers
- Rapier 48*i* Layer 3 Switch
- A40/SC-00, A40/MT-00, A41/SC-00, and A41/MT-00 100Base-FX uplink modules for Rapier G6 and G6F switches, and all Rapier *i* Series switches
- A42/GBIC uplink module for all Rapier and Rapier *i* Series switches, and AR800 Series modular switching routers

## New Features

Table 1 summarises the new features and enhancements in Software Release 2.4.1 by product. Each new feature and enhancement is described in detail in the following sections.

**Table 1: New features and enhancements in Software Release 2.4.1 by product series.**

	AR300	AR400	AR700	Rapier	Rapier <i>i</i>	AR800
<b>Major Features</b>						
IPsec support for IPv6	✓	✓	✓	✓	✓	✓
Encryption key support for IPv6	✓	✓	✓	✓	✓	✓
IPv6 6-to-4 dynamic tunnelling	✓	✓	✓	✓	✓	✓
IGMP Proxy	✓	✓	✓	✓	✓	✓
BGP-4 triggers		✓	✓	✓	✓	✓
IPv6 multicasting	✓	✓	✓			
AppleTalk filtering	✓	✓	✓	✓	✓	✓
Graphical User Interface (GUI)				✓	✓	
Multicast VLAN Registration (MVR)				✓	✓	✓
Classifiers				✓	✓	✓
Hardware packet filters				✓	✓	✓
Quality of Service (QoS) support					✓	
Access Control Lists (ACLs)					✓	
Overlapping VLANs/STPs					✓	
<b>Minor Improvements</b>						
Configurable MRU and MAXLINKS options	✓	✓	✓	✓	✓	✓
IPv6 link-local addresses	✓	✓	✓	✓	✓	✓
IPv6 Neighbour Discovery enhancements	✓	✓	✓	✓	✓	✓
IPv6 Path MTU Discovery	✓	✓	✓	✓	✓	✓
IPv6 Poison Reverse on RIP	✓	✓	✓	✓	✓	✓
Setting IGMP thresholds	✓	✓	✓	✓	✓	✓
COPY command	✓	✓	✓	✓	✓	✓
3DES Outer CBC encryption mode	✓	✓	✓	✓	✓	✓
IP route filtering	✓	✓	✓	✓	✓	✓
DHCP Extended Client ID	✓	✓	✓	✓	✓	✓
OSPF/RIP static export control parameter	✓	✓	✓	✓	✓	✓
Ping Timeout	✓	✓	✓	✓	✓	✓
TCP listen ports closed by default	✓	✓	✓	✓	✓	✓
Layer 3 Ageing Timer				✓	✓	✓

**Table 1: New features and enhancements in Software Release 2.4.1 by product series.**

	AR300	AR400	AR700	Rapier	Rapier i	AR800
Ingress filtering				✓	✓	✓
Protocol Type parameter for Layer 3 hardware filters				✓	✓	✓
Enhancements to IP multicast switching					✓	



*In this document, the term router is used both to refer specifically to AR router products, and also to refer to generic devices (switches or routers) providing routing functionality such as IP, OSPF, AppleTalk or multicast routing.*

## Graphical User Interface (GUI)

A wide range of switch functionality can be configured, managed and monitored through the switch's Graphical User Interface (GUI). For a description of how to access the GUI, see your switch's Quick Install Guide, or the *Operation* chapter of the *Software Reference*.

For context-sensitive help, click the GUI Assistant button on any GUI page.

The following software features can be configured through the GUI:

- System identity, time configuration and NTP, triggers and SNMP
- Port configuration, including mirroring, trunking and packet storm limiting
- VLAN configuration, spanning trees, and GVRP
- Internet Protocol: DNS, interfaces, static routes, RIP, multicasting and OSPF
- Novell IPX
- Classifiers, Quality of Service and hardware filters

The following switch functionality can be managed through the GUI:

- User accounts
- File creation, editing, backup and restoring
- Boot configuration
- Software release and feature licences
- Software upgrades

The GUI also contains an interface to the switch's command line interface, allowing any switch command to be entered.

The switch automatically generates log messages. Messages can be viewed through the GUI, and filters can be set up to determine where messages are saved to and which messages are saved.

The following switch functionality can be monitored through the GUI:

- System status and hardware details
- Port statistics
- Layer 2 forwarding database
- ARP table
- IP route table

The following counters can be viewed through the GUI:

- Port
- STP and GVRP
- IP, ICMP and UDP, RIP and routes, and IPX

Contents of the switch's file system, and associated counters, can also be viewed.

## COPY Command

A new command, COPY, can be used to copy any text file within FLASH memory, within NVS, and between FLASH and NVS:

```
COPY [device:] filename1.ext [device:] filename2.ext
```

where *device* is the physical location of the file ("flash" or "nvs"). The default is "flash". Only text files (files with an extension of "txt", "cfg", "scp", "hlp", "htm", "spa" or "mds") can be copied. The original file and the copy must have the same extension.

## Enhancements to Switching

The following enhancements have been made to basic switching functionality in Software Release 2.4.1:

- The INFILTERING parameter defaults to OFF
- Configurable aging timer for Layer 3 forwarding database entries

## Ingress Filtering

The default for the INFILTERING parameter of the SET SWITCH PORT command is now OFF:

```
SET SWITCH PORT={port-list|ALL} [ACCEPTABLE={ALL|VLAN}]
[BCLIMIT={NONE|limit}] [DESCRIPTION=description]
[DLFLIMIT={NONE|limit}]
[EGRESSLIMIT={NONE|DEFAULT|0|1000..127000|8..1016}]
[INFILTERING={OFF|ON}]
[INGRESSLIMIT={NONE|DEFAULT|0|64..127000|8..1016}]
[LEARN={NONE|0|1..256}]
[INTRUSIONACTION={DISABLE|DISCARD|TRAP}]
[MCLIMIT={NONE|limit}] [MIRROR={BOTH|NONE|RX|TX}]
[MODE={AUTONEGOTIATE|MASTER|SLAVE}]
[MULTICASTMODE={A|B|C}] [SPEED={AUTONEGOTIATE|10MHALF|
10MFULL|10MHAUTO|10MFAUTO|100MHALF|100MFULL|100MHAUTO|
100MFAUTO|1000MHALF|1000MFULL|1000MHAUTO|1000MFAUTO}]
```

The `INFILTERING` parameter specifies whether or not Ingress Filtering is enabled. If `ON` is specified, any frame received on a specified port is only admitted if the port belongs to the VLAN with which the frame is associated. Conversely, any frame received on the port is discarded if the port does not belong to the VLAN with which the frame is associated. Untagged frames admitted by the `ACCEPTABLE` parameter are admitted, since they have the numerical VLAN Identifier (VID) of the VLAN for which the port is an untagged member. If `OFF` is specified, Ingress Filtering is disabled, and no frames are discarded by this part of the Ingress Rules.

### Layer 3 Ageing Timer

A new command has been added to enable the threshold of the ageing timer for dynamic entries in the Layer 3 forwarding database to be set:

```
SET SWITCH L3AGEINGTIMER=30..43200
```

This command sets the threshold value, in seconds, of the ageing timer for dynamic entries in the Layer 3 forwarding database. After a cycle of this timer, entries that were not used during this cycle will have their hit bit reset to zero, but will remain in the table. After the next cycle of the timer, entries with their hit bit still set to zero will be deleted. Therefore, entries in the table are only deleted if they are unused during two consecutive cycles of the timer. The default is 900.

This command can only be executed if the hardware forwarding entry ageing timer is enabled using the `ENABLE SWITCH AGEINGTIMER` command. This ageing timer is enabled by default.

### Generic Packet Classifiers

A generic packet classifier categorises incoming packets into data flows based on a wide range of specific characteristics, such as destination IP address, and general characteristics, such as IP protocol. A classifier only sorts packets; it does not process them. The sorted packet flows can be processed by QoS (Quality of Service) or hardware filters, as required.

To define a generic packet classifier, use the command:

```
CREATE CLASSIFIER=rule-id [VLAN={vlannname|1..4094|ANY}]
  [EPORT=portnum] [ETHFORMAT={802.2|802.2-TAGGED|
802.2-UNTAGGED|ETHII|ETHII-TAGGED|ETHII-UNTAGGED|
NETWARERAW|SNAP|ANY}] [IPDADDR={ipadmask|ANY}]
  [IPSADDR={ipadmask|ANY}] [IPDSCP={0..63|ANY}]
  [IPOINT=portnum] [IPPROTOCOL={TCP|UDP|ICMP|IGMP|
ipprotocolnum}] [IPTOS={0..7|ANY}] [IPXDADDR={ipxad|ANY}]
  [IPXSOCKET={NCP|SAP|RIP|NNB|DIAG|NLSP|IPXWAN|
ipxsocketnum|ANY}] [IPXSOCKET={NCP|SAP|RIP|NNB|DIAG|
NLSP|IPXWAN|ipxsocketnum|ANY}] [IPXPACKET={NLSP|RIP|SAP|
SPX|NCP|NETBIOS|ipxpacketnum|ANY}] [MACDADDR={macadd|
ANY}] [MACSADDR={macadd|ANY}] [MATCH1=hhh] [MASK1=hhh]
[OFFSET1=0..62] [MATCH2=hhh] [MASK2=hhh] [OFFSET2=0..62]
[MATCH3=hhh] [MASK3=hhh] [OFFSET3=0..62]
[PROTOCOL={protocoltype|ANY}] [TCPDPORT={portid|ANY}]
[TCPSPORT={portid|ANY}] [TCPFLAGS={{URG|ACK|RST|SYN|
FIN}|...}|ANY}] [UDPDPORT={portid|ANY}]
[UDPSPORT={portid|ANY}]
```

To delete a generic packet classifier, use the command:

```
DESTROY CLASSIFIER={rulelist|ALL}
```

To modify an existing generic packet classifier, use the command:

```
SET CLASSIFIER=rule-id [VLAN={vlaname|1..4094|ANY}]
[EPORT=portnum] [ETHFORMAT={802.2|802.2-TAGGED|
802.2-UNTAGGED|ETHII|ETHII-TAGGED|ETHII-UNTAGGED|
NETWARERAW|SNAP|ANY}] [IPDADDR={ipaddmask|ANY}]
[IPSADDR={ipaddmask|ANY}] [IPDSCP={0..63|ANY}
Iport=portnum] [IPPROTOCOL={TCP|UDP|ICMP|IGMP|
ipprotocolnum}] [IPTOS={0..7|ANY}] [IPXDADDR={ipxadd|ANY}]
[IPXD SOCKET={NCP|SAP|RIP|NNB|DIAG|NLSP|IPXWAN|
ipxsocketnum|ANY}] [IPXS SOCKET={NCP|SAP|RIP|NNB|DIAG|
NLSP|IPXWAN|ipxsocketnum|ANY}] IPXPACKET={NLSP|RIP|SAP|
SPX|NCP|NETBIOS|ipxpacketnum|ANY}] [MACDADDR={macadd|
ANY}] [MACSADDR={macadd|ANY}] [MATCH1=hhh] [MASK1=hhh]
[OFFSET1=0..62] [MATCH2=hhh] [MASK2=hhh] [OFFSET2=0..62]
[MATCH3=hhh] [MASK3=hhh] [OFFSET3=0..62]
[PROTOCOL={protocoltype|ANY}] [TCPDPORT={portid|ANY}]
[TCPSPORT={portid|ANY}] [TCPFLAGS={URG|ACK|RST|SYN|
FIN|[,...]|ANY}] [UDPDPOR T={portid|ANY}]
[UDPSPORT={portid|ANY}]
```

To display information about generic packet classifiers, use the command:

```
SHOW CLASSIFIER[=rulelist|ALL] [VLAN={vlaname|1..4094|
ANY}] [EPORT=portnum] [ETHFORMAT={802.2|802.2-TAGGED|
802.2-UNTAGGED|ETHII|ETHII-TAGGED|ETHII-UNTAGGED|
NETWARERAW|SNAP|ANY}] [IPDADDR={ipaddmask|ANY}]
[IPSADDR={ipaddmask|ANY}] [IPDSCP={0..63|ANY}
Iport=portnum] [IPPROTOCOL={TCP|UDP|ICMP|IGMP|
ipprotocolnum|ANY|NONTCPUDP}] [IPTOS={0..7|ANY}]
[IPXDADDR={ipxadd|ANY}] [IPXD SOCKET={NCP|SAP|RIP|NNB|
DIAG|NLSP|IPXWAN|ipxsocketnum|ANY}] [IPXS SOCKET={NCP|SAP|
RIP|NNB|DIAG|NLSP|IPXWAN|ipxsocketnum|ANY}]
IPXPACKET={NLSP|RIP|SAP|SPX|NCP|NETBIOS|ipxpacketnum|
ANY}] [MACDADDR={macadd|ANY}] [MACSADDR={macadd|ANY}]
[MATCH1=hhh] [MASK1=hhh] [OFFSET1=0..62] [MATCH2=hhh]
[MASK2=hhh] [OFFSET2=0..62] [MATCH3=hhh] [MASK3=hhh]
[OFFSET3=0..62] [PROTOCOL={protocoltype|ANY}]
[TCPDPORT={portid|ANY}] [TCPSPORT={portid|ANY}]
[TCPFLAGS={URG|ACK|RST|SYN|FIN|[,...]|ANY}]
[UDPDPOR T={portid|ANY}] [UDPSPORT={portid|ANY}]
```

Once a classifier has been defined, it must be configured for use by QoS or hardware filters. For example, to add a hardware-based packet filter that uses a classifier, use the command:

```
ADD SWITCH HWFILTER CLASSIFIER=classifier-list
[ACTION={SETPRIORITY|SENDCOS|SETTOS|DENY|SENDEPORT|
SENDMIRROR|MOVEPRIOTOTOS|MOVETOSTOPRIO|SETIPDSCP|
SENDNONUNICASTTOPORT|NODROP|FORWARD|[,...]}]
[NEWIPDSCP=dscp-value] [NEWTOS=0..7]
[NOMATCHACTION={SETPRIORITY|SENDCOS|SETTOS|DENY|
SENDEPORT|SENDMIRROR|MOVEPRIOTOTOS|MOVETOSTOPRIO|
SETIPDSCP|SENDNONUNICASTTOPORT|FORWARD|[,...]}]
[NOMATCHDSCP=dscp-value] [NOMATCHPORT=port-number]
[NOMATCHPRIORITY=0..7] [NOMATCHTOS=0..7]
[PORT=port-number] [PRIORITY=0..7]
```



*Not all classifier parameters and options are supported on all hardware platforms.*

## Hardware Packet Filters

Switch hardware can be configured to discard, forward, mirror, or change the priority of packets matching specified criteria at wirespeed. For Rapier *i* Series switches, filters can also be configured to provide a range of Quality of Service (QoS) controls, including changing the DSCP byte, and actions can be specified for packets which match the ingress and egress ports of the filter (if set), but do not match the filter's other parameters.

Two sets of commands are available, one based on existing Layer 3 hardware filter matches and entries, and one based on generic packet classifiers added in Software Release 2.4.1. These two filter types cannot be used together. Classifier-based packet filters provide more flexibility and enhanced options, and are recommended over Layer 3 hardware filters.

## Classifier-based Packet Filters

The switch hardware can be configured through entries in the packet classifier to copy, drop, forward, and associate QoS attributes to Layer 3 packets that match the criteria set using the classifier.

Every packet passing through the switch is matched against a series of classification tables by the packet classifier. Packets can be classified according to:

- Packet type
- Physical source/destination port
- Layer 3 protocol
- Source/destination IP address
- Destination IPX address
- Layer 4 protocol (for example: TCP/UDP/Socket number)
- Layer 4 source/destination ports
- Any 16-bit word in the first 64 bytes of a packet

Hardware-based packet filters can be configured to take action based on the results of the classification tables. These actions are:

- Discard the packet
- Forward the packet
- Send the packet to the mirror port
- Forward the packet to a specified egress port, for unicast packets
- Send the packet to a Class of Service queue
- Replace the packet's 802.1p priority

For Rapier *i* Series switches, the filter can also perform the following Quality of Service actions:

- Replace the packet's IP TOS value and/or the IP DSCP value
- Direct non-unicast packets that were scheduled to be dropped or sent to the CPU to a specified port
- Forward packets that were marked to be dropped. This option allows bandwidth limiting to be overridden for particular packets.

For Rapiet *i* Series switches, all actions are also available on packets which match the ingress and egress ports of the classifier (if either or both are set), but do not match the classifier's other parameters.

A classifier-based packet filter comprises a single classifier entry. A number of filters can be created at one time with the same action, by specifying a list of classifiers, but each classifier will be contained in a single filter. The number of packet filters supported by the switch is determined by the switch model and how different each filter is.

To enable and disable classifier-based hardware filtering, use the commands:

```
ENABLE SWITCH HWFILTER
DISABLE SWITCH HWFILTER
```

To add hardware-based packet filters to the switch, use the command:

```
ADD SWITCH HWFILTER CLASSIFIER=classifier-list
[ACTION={SETPRIORITY|SENDCOS|SETTOS|DENY|SENDEPORT|
SENDMIRROR|MOVEPRIOTOTOS|MOVETOSTOPRIO|SETIPDSCP|
SENDNONUNICASTTOPORT|NODROP|FORWARD} [, ...]]
[NEWIPDSCP=dscp-value] [NEWTOS=0..7]
[NOMATCHACTION={SETPRIORITY|SENDCOS|SETTOS|DENY|
SENDEPORT|SENDMIRROR|MOVEPRIOTOTOS|MOVETOSTOPRIO|
SETIPDSCP|SENDNONUNICASTTOPORT|FORWARD} [, ...]]
[NOMATCHDSCP=dscp-value] [NOMATCHPORT=port-number]
[NOMATCHPRIORITY=0..7] [NOMATCHTOS=0..7]
[PORT=port-number] [PRIORITY=0..7]
```

To delete one or more hardware-based packet filters from the switch, use the command:

```
DELETE SWITCH HWFILTER CLASSIFIER=classifier-list
```

To display information about hardware-based packet filters, use the command:

```
SHOW SWITCH HWFILTER [CLASSIFIER=classifier-list]
```

## Protocol Type Parameter for Layer 3 Hardware Filters

Layer 3 hardware filters can now filter on protocol type. First, specify the type in the filter match, using one of the commands:

```
ADD SWITCH L3FILTER MATCH={DIPADDR|IPDSCP|PROTOCOL|SIPADDR|
TCPACK|TCPFIN|TCPDPORT|TCPSPORT|TCPSYN|TOS|TTL|UDPDPORT|
UDPSPORT} [, ...] [TYPE={802|ETHII|SNAP}] [other-options...]
SET SWITCH L3FILTER=filter-id MATCH={DIPADDR|IPDSCP|PROTOCOL|
SIPADDR|TCPACK|TCPFIN|TCPDPORT|TCPSPORT|TCPSYN|TOS|TTL|
UDPDPORT|UDPSPORT} [, ...] [TYPE={802|ETHII|SNAP}]
[other-options...]
```

The TYPE parameter specifies the format of the protocol type. This parameter may be used with the EMPORT and IMPORT parameters, but not with the other packet matching criteria. When other criteria are used there is an implicit match to an IP protocol Ethernet type II packet. If 802 is specified then the match will be on the 2-byte DSAP/SSAP field of an 802.3 packet. If ETHII is specified then the match will be on the 2-byte type field of an Ethernet type II packet. If SNAP is specified then the match will be on the 5-byte variable part of the identifier field of a SNAP packet (SNAP identifiers have the format *aa-aa-03-xx-xx-xx-xx-xx*).

Then specify the type in the filter entry, using one of the commands:

```
ADD SWITCH L3FILTER=filter-id ENTRY [TYPE=protocol-type]
[other options]

SET SWITCH L3FILTER=filter-id ENTRY=entry-id
[TYPE=protocol-type] [other options]
```

where *protocol-type* is a valid protocol type number. A protocol type number is 2 bytes for Ethernet type II and 802.3 (DSAP/SSAP) encapsulation, or 5 bytes for SNAP encapsulation, and is specified in hexadecimal.

The TYPE parameter specifies a protocol type number to match. The number is entered in hexadecimal, e.g. 0800 for an Ethernet type II IP packet. This parameter may not be used with any other packet field matching criteria, nor may it be used with the SETTOS action. With all other packet matching criteria there is an implicit match to an IP protocol Ethernet type II packet.

## Quality of Service (QoS)

Quality of Service (QoS) enables the switch to manage traffic queues, prioritise traffic, and limit bandwidth. The switch's QoS features comprise a tool set for performing a range of QoS applications, including configuration of a Differentiated Services (DiffServ) domain.

To configure QoS:

1. Create Classifiers, to sort packets into traffic flows.
2. Create Flow Groups, using the command:

```
CREATE QOS FLOWGROUP=flowgrouplist
[DESCRIPTION=description] [MARKVALUE={dscpvalue|NONE}]
[PRIORITY={priority|NONE}]
[REMARKPRIORITY={YES|NO|ON|OFF|TRUE|FALSE}]
```

Flow groups are groups of classifiers, and group together similar traffic flows. QoS prioritisation and/or DiffServ Code Point (DSCP) replacement can be applied to flow groups.

3. Add the classifiers to the flow groups, using the command:

```
ADD QOS FLOWGROUP=flowgroupid CLASSIFIER=classifierlist
```

4. Create Traffic Classes, using the command:

```
CREATE QOS TRAFFICCLASS=idlist [DESCRIPTION=description]
[EXCEEDACTION={DROP|REMARK}]
[EXCEEDREMARKVALUE=dscpvalue]
[MARKVALUE={dscpvalue|NONE}] [MAXBANDWIDTH=bandwidth]
[PRIORITY={priority|NONE}]
[REMARKPRIORITY={YES|NO|ON|OFF|TRUE|FALSE}]
```

Traffic classes are groups of flow groups and are central to QoS. QoS bandwidth limiting, prioritisation and/or DSCP replacement is generally applied to traffic classes.

5. Add the flow groups to the traffic classes, using the command:

```
ADD QOS TRAFFICCLASS=tcid FLOWGROUP=flowgrouplist
```

Each flow group can only be assigned to one traffic class.

6. Create Policies, using the command:

```
CREATE QOS POLICY=idlist [DESCRIPTION=description]
[INDSCPOVERWRITE=dscpvalue|NONE]
[REMARKINDSCP=ZERO|ALL|NONE]
```

Policies are groups of traffic classes. A policy defines a complete QoS solution for a port or group of ports.

7. Add the traffic classes to the policies, using the command:

```
ADD QOS POLICY=id TRAFFICCLASS=tcidlist
```

Each traffic class can only be assigned to one policy.

8. Attach ports to the policies, using the command:

```
SET QOS PORT={portlist|ALL} POLICY=id
```

Each port can only have one policy. QoS controls are applied to traffic egressing ports. Therefore, to control a particular type of traffic, each port that type of traffic egresses must have an appropriate QoS policy attached. The same policy can probably be used on all ports, unless the traffic is classified by a port-specific classifier, such as destination IP address.

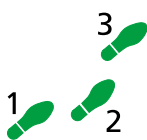
Other QoS functionality includes commands to:

- Alter the assignment of VLAN tag precedence field to Class of Service (and therefore egress queue priority)
- Replace the VLAN tag User Priority on egress
- Configure Weighted Round Robin Queueing with bounded delay
- Modify and destroy existing QoS controls
- Display information about QoS

## Access Control Lists (ACLs)

On Rapier *i* Series switches, classifiers and hardware packet filters can be configured to provide Access Control List functionality.

For example, to allow WWW servers in the 192.168.10.0 subnet to be accessed only from the 192.168.20.0 subnet:



1. **Create a classifier to match all WWW traffic to the subnet**

Create a classifier to match all WWW traffic to the 192.168.10.0 subnet:

```
CREATE CLASSIFIER=1 IPDADDR=192.168.10.0/24 TCPDPORT=80
```

2. **Create a hardware packet filter to deny this traffic**

```
ADD SWITCH HWFILTER CLASSIFIER=1 ACTION=DENY
```

3. **Create a classifier to match the subset of this traffic that is to be allowed**

Create a classifier to match WWW traffic from the 192.168.20.0 subnet to the 192.168.10.0 subnet:

```
CREATE CLASSIFIER=2 IPDADDR=192.168.10.0/24  
IPSADDR=192.168.20.0/24 TCPDPORT=80
```

4. **Create a hardware packet filter to allow this traffic**

This filter must be created last, so it will be the first filter the switch processes:

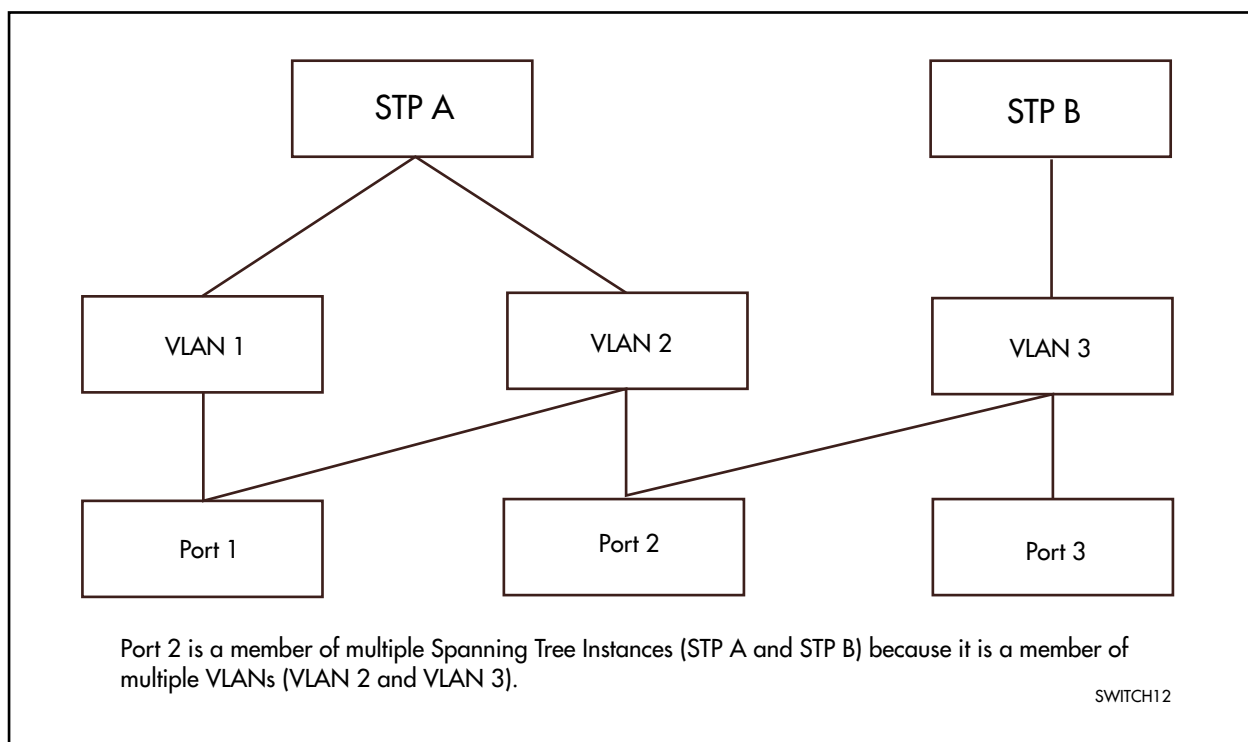
```
ADD SWITCH HWFILTER CLASSIFIER=2 ACTION=NODROP
```

## Overlapping VLANs/STPs

The Spanning Tree Protocol (STP) is a link management protocol used to dynamically determine the best path from source to destination and provide path redundancy. An algorithm is used to create a hierarchical “tree” that “spans” the entire network. There can be multiple paths from source to destination. However, for an Ethernet network to function correctly there can only be one active path. STP ensures that the most efficient path is used and prevents the condition known as a bridge loop. If a link in the network fails, STP reconfigures the spanning tree topology, activating a standby path to reestablish the link.

The Rapier *i* Series switch allows a port to belong to more than one Spanning Tree instance when the port is a member of more than one VLAN and those VLANs belong to different STPs.

**Figure 1: Port membership of VLANs which belong to different spanning tree instances (on the Rapier *i* Series switch only).**



The ADD STP VLAN command has been enhanced to allow a port to belong to more than one STP if the port is a member of two or more VLANs which belong to different STPs. To add a VLAN to an STP, use the command:

```
ADD STP=stp-name VLAN={vlan-name|2..4094}
```

The ADD VLAN PORT and DELETE VLAN PORT commands have been enhanced to allow a port belong to multiple STPs if the port is a member of more than one VLAN. If the port being added or deleted from the VLAN also belongs to another STP, through concurrent membership of another VLAN, it will not be removed from that VLAN or STP.

To add or delete ports from a VLAN, use the commands:

```
ADD VLAN={vlan-name|1..4094} PORT={port-list|ALL}
[FRAME={TAGGED|UNTAGGED}]
```

```
DELETE VLAN={vlan-name|1..4094} PORT={port-list|ALL}
```

Up to 255 STPs can be configured. To create a Spanning Tree Protocol entity with a unique name, use the command:

```
CREATE STP=stp-name
```

A port can belong to more than one STP after deletion. If a port belongs to multiple VLANs in the same STP, the port will remain a member of this STP if a VLAN it was a member of is deleted from the STP and returned to the default STP. To delete one, or all, VLANs from the specified STP, and return the VLANs to the default STP, use the command:

```
DELETE STP=stp-name VLAN={vlan-name|2..4094|ALL}
```

To enable or disable operation of the Spanning Tree Algorithm on specified ports, use the commands:

```
ENABLE STP[={stp-name|ALL}] PORT={port-list|ALL}
```

```
DISABLE STP[={stp-name|ALL}] PORT={port-list|ALL}
```

The STP parameter specifies the STP for which the ports will be enabled or disabled. If an STP is not specified, the default is ALL. If an STP is specified, and the port or ports belong to more than one STP, they will be enabled or disabled only on the specified STP. The state of the ports in the other STPs remains unchanged.

To set various parameters used by the Spanning Tree Algorithm for specified ports in an STP, use the command:

```
SET STP[={stp-name|ALL}] PORT={port-list|ALL}  
[PATHCOST=1..1000000] [PORTPRIORITY=0..255]
```

```
SET STP[={stp-name|ALL}] PORT={port-list|ALL} DEFAULT
```

The STP parameter specifies the STP. If an STP is not specified, the default is ALL.

To enable or disable STP debugging options for an STP or ports, use the command:

```
ENABLE STP={stp-name|ALL} DEBUG={MSG|PKT|STATE|ALL}  
[OUTPUT=CONSOLE] [TIMEOUT={1..4000000000|NONE}]
```

```
ENABLE STP={stp-name|ALL} DEBUG={MSG|PKT|STATE|ALL}  
PORT={port-list|ALL} [OUTPUT=CONSOLE]  
[TIMEOUT={1..4000000000|NONE}]
```

```
DISABLE STP={stp-name|ALL} DEBUG={MSG|PKT|STATE|ALL}  
PORT={port-list|ALL}
```

```
DISABLE STP DEBUG={MSG|PKT|STATE|ALL} PORT={port-list|ALL}
```

The PORT parameter specifies the ports on which the debugging mode is enabled or disabled. If an STP is not specified, or ALL is specified as the STP name, then the debugging mode will be enabled or disabled for the listed ports in all the STP's which have the listed ports as members. Otherwise, only the specified STP will have the debugging mode enabled or disabled on the listed ports. The DEBUG parameter must be specified before the PORT parameter.

The SHOW STP DEBUG command has been enhanced to display information about a specified STP. To display the debugging modes enabled on each port, use the command:

```
SHOW STP[={stp-name|ALL}] DEBUG
```

If an STP name is not specified, the default is ALL. A new field, *STP Name*, has been added (See Figure 2 on page 14 and Table 2 on page 14).

**Figure 2: Example output from the SHOW STP DEBUG command.**

STP Name	Port	Enabled Debug Modes	Output	Timeout
-----				
default	Port1	MSG, PKT, STATE	Console (16)	NONE
	Port2	STATE	Console (16)	12345
	Port3	None		
-----				
Admin	Port1	MSG, PKT, STATE	TTY (12)	100
-----				

**Table 2: New parameter displayed in the output of the SHOW STP DEBUG command.**

Parameter	Meaning
STP name	Name of the STP instance.

The SHOW STP PORT command has been enhanced to display Spanning Tree Protocol port information for the specified ports, or all ports. To display Spanning Tree Protocol port information, use the command:

```
SHOW STP [= {stp-name|ALL}] PORT [= {port-list|ALL}]
```

If an STP name is not specified, the default is ALL. A new field, *VLAN Membership*, has been added (See Figure 3 on page 15 and Table 3 on page 15).

**Figure 3: Example output from the SHOW STP PORT command**

```

STP Port Information
-----
STP Name ..... default
STP Status ..... ON
Port ..... 1
  State ..... Forwarding
  Port Priority ..... 128
  Port Identifier ..... 8001
  Pathcost ..... 19
  Designated Root ..... 32768 : 00-00-cd-00-45-c7
  Designated Cost ..... 0
  Designated Bridge ... 32768 : 00-00-cd-00-a9-a5
  Designated Port ..... 8001
  VLAN membership ..... 2

STP Name ..... Admin
STP Status ..... OFF
Port ..... 1
  State ..... Disabled
  Port Priority ..... 128
  Port Identifier ..... 8001
  Pathcost ..... 101
  Designated Root ..... 32768 : 00-00-cd-00-45-c7
  Designated Cost ..... 0
  Designated Bridge ... 32768 : 00-00-cd-00-a9-a5
  Designated Port ..... 8001
  VLAN membership ..... 1
-----

```

**Table 3: New parameter displayed in the output of the SHOW STP PORT command .**

Parameter	Meaning
VLAN membership	The number of VLANs the port is a member of within this STP instance.

To set the port parameters of a GARP application, use the command:

```

SET GARP=GVRP PORT={port-list|ALL} [MODE={NONE|NORMAL}]
  [SHOWDEBUG={ON|OFF}] [STP={stp-name|ALL}]

```

If the port list supplied does not match all the STP instances perfectly, the command will still succeed as a whole. The STP parameter specifies which GVRP instance is affected by this command. Each STP has its own instance of GVRP. The default ALL. Because a port can belong to more than one STP instance a port can belong to more than one instance of a GVRP. If the STP parameter ALL is specified for a port then all GVRP instances associated with the port will be affected.

By specifying the SHOWDEBUG parameter on the SET GARP PORT command the output of debugging can be switched on or off on a per-port, per STP instance basis, using the commands:

```

ENABLE GARP=GVRP DEBUG={MSG|PKT|STATE|ALL} [OUTPUT=CONSOLE]
  [TIMEOUT={NONE|1..400000000}] [STP={stp-name|ALL}]

DISABLE GARP=GVRP DEBUG={MSG|PKT|STATE|ALL} [STP={stp-name|
  ALL}]

```

## Configurable MRU and MAXLINKS options

The MRU (Maximum Receive Unit) option can now be disabled for PPP links. When a Point-to-Point (PPP) link is brought up, a Link Control Protocol (LCP) link request may include an Maximum Receive Unit (MRU) option, indicating to the remote endpoint the largest packet size that the sender of the option can receive from the endpoint. Third-party products may, on the receipt of this option, terminate the link. To allow interoperability with these products it may be necessary to disable transmission of the MRU option.

Point-to-Point (PPP) links can be configured to disable the transmission of the MRU option in their LCP negotiation. If a command explicitly disables the MRU option for a PPP interface or template, all subsequent requests on that interface will omit the MRU option. The MRU option is enabled by default.

To disable or enable the MRU option when creating a PPP interface or PPP template, use the commands:

```
CREATE PPP=ppp-interface OVER=physical-interface
      [MRU={ON|OFF}] [other-options...]

CREATE PPP TEMPLATE=template [MRU={ON|OFF}]
      [other-options...]
```

To modify the MRU option on an existing PPP interface or PPP template, use the commands:

```
SET PPP [=ppp-interface] [MRU={ON|OFF}] [other-options...]

SET PPP TEMPLATE=template [MRU={ON|OFF}] [other-options...]
```

The MAXLINKS parameter limits the number of dynamic PPP links that can be added to a multilink bundle as the result of incoming calls. The default value is 2. The MAXLINKS parameter is most applicable to ISDN links where the remote (receiving) end of the PPP link has a limited number of data channels (e.g. a single PRI interface). In this situation, MAXLINKS can be used to prevent a single PPP multilink bundle from monopolising all the B channels.

To set the maximum number of dynamic PPP links in a multilink bundle, create a PPP template using the command:

```
CREATE PPP TEMPLATE=template [MAXLINKS=1..64]
      [other-options...]
```

To modify the maximum number of dynamic links per multilink bundle on an existing PPP template, use the command:

```
SET PPP TEMPLATE=template [MAXLINKS=1..64] [other-options...]
```

To modify the maximum number of dynamic links per multilink bundle on an existing dynamic PPP interface, use the commands:

```
SET PPP [=ppp-interface] [MAXLINKS=1..64] [other-options...]
```

The MAXLINKS parameter is best used in conjunction with BAP (Bandwidth Allocation Protocol), to prevent link flapping and call charges. Three scenarios illustrate the possibilities and ramifications:

### ■ Static PPP interface without BAP:

The initiating (static) end of the data link has a statically defined PPP interface with multiple PPP links (e.g. NUMBER=5) in a multilink bundle. When the PPP interface is created (or after a reboot) PPP will attempt to open all the links in the multilink bundle and keep them open. If a link is

closed for any reason, PPP will attempt to re-open it immediately. If the remote (dynamic) end of the link has MAXLINKS set to a lower value (e.g. MAXLINKS=3), it will reject the additional links. Because the PPP link is brought up fully before it is checked, links may be brought up and torn down repeatedly causing link flapping and incurring call charges for each call made attempting to open the additional links.

■ **On-demand PPP interface without BAP:**

The initiating (static) end of the data link has a statically defined PPP interface with multiple dial-on-demand PPP links (e.g. NUMBER=5) in a multilink bundle. When there is data for transmission, PPP will attempt to open all the links in the multilink bundle. If a link is closed for any reason, PPP will not attempt to re-open it unless there is data to transmit. As long as at least one link in the multilink bundle is successfully opened, data transmission will continue. If the remote (dynamic) end of the link has MAXLINKS set to a lower value (e.g. MAXLINKS=3), it will reject the additional links. Because the PPP link is brought up fully before it is checked, the links will be brought up and torn down. However, because the initiating end does not attempt to retry the failed links, link flapping and call charges are minimal.

■ **BAP-controlled PPP interface:**

The initiating (static) end of the data link has a statically defined PPP interface with multiple dial-on-demand PPP links (e.g. NUMBER=5) in a multilink bundle, with BAP enabled. When there is data for transmission, BAP will control the opening of the links in the multilink bundle. If the remote (dynamic) end of the link has MAXLINKS set to a lower value (e.g. MAXLINKS=3), it will reject the BAP requests to open additional links, before the links are opened. Link flapping and call charges are eliminated.

A log message is generated whenever the MAXLINKS setting causes a PPP link to be torn down.

## IP Route Filtering

The DIRECTION and PROTOCOL parameters of the ADD IP ROUTE FILTER and SET IP ROUTE FILTER commands have been enhanced:

```
ADD IP ROUTE FILTER [=filter-id] IP=ipadd MASK=ipadd
ACTION={ INCLUDE | EXCLUDE | SWITCH }
[DIRECTION={ RECEIVE | SEND | BOTH } ] [INTERFACE=interface]
[NEXTHOP=ipadd] [POLICY=0..7]
[PROTOCOL={ ANY | EGP | OSPF | RIP | STATIC | INTERFACE } ]

SET IP ROUTE FILTER=filter-id [IP=ipadd] [MASK=ipadd]
[ACTION={ INCLUDE | EXCLUDE | SWITCH } ]
[DIRECTION={ RECEIVE | SEND | BOTH } ] [INTERFACE=interface]
[NEXTHOP=ipadd] [POLICY=0..7]
[PROTOCOL={ ANY | EGP | INTERFACE | OSPF | RIP | STATIC } ]
```

The DIRECTION parameter specifies whether to filter the route when receiving it or when sending it. If RECEIVE or BOTH is specified, the filter affects the addition of routes to the IP routing table. If SEND or BOTH is specified, the filter affects the selection of routes from the IP routing table to be advertised or propagated.

The PROTOCOL parameter specifies the routing protocol to which the filter applies. The default is ANY. When DIRECTION is RECEIVE, the PROTOCOL parameter specifies the routing protocol receiving the route information. If the

DIRECTION is SEND, then the PROTOCOL parameter specifies the routing protocol advertising the routes.

IP route filters apply to the interaction between the routing protocol and the IP routing table, not to the reception and transmission of routing messages by the routing protocol. Messages sent from the routing protocol are affected if they are derived from the IP routing table, which is true in most situations, including RIP, OSPF external LSAs and OSPF summary Link State Advertisements (LSAs). Some types of OSPF LSAs, such as intra-area LSAs, are not filtered by route filters because they are not based on the local IP route table, and to meet OSPF design requirements of LSA propagation within areas.

The immediate effect of adding an IP route filter with a DIRECTION of RECEIVE or BOTH is that any received route advertisements which match the filter will not result in a new entry in the local IP route table. However, any routes already existing in the IP route table will not be deleted, even if they match the route filter. Therefore, when dynamically adding an IP route filter, it may be necessary to manually delete from the IP routing table any unwanted routes that already exist.

Route filters defined with a DIRECTION of RECEIVE or BOTH may also have a subsequent effect on outgoing OSPF Link State Advertisements (LSAs). The behaviour will depend on the router's OSPF role. On an Area Border Router, a matching RECEIVE direction filter will have the subsequent effect that no summary LSA will be advertised into another area, because summary LSA messages are derived from the filtered IP route table. In contrast, other OSPF area members derive their messages from their local OSPF configuration and their received LSA messages. Note that a OSPF design requirement is that LSA messages must not be filtered within an OSPF area.

How the OSPF protocol is implemented determines how the route filter affects OSPF Link State Advertisement (LSA). A route filter with a DIRECTION of SEND or BOTH will only filter matching routes regarded as Autonomous System (AS) external routes by OSPF. Also, the INTERFACE parameter is ignored and all interfaces are treated the same.

To filter OSPF inter-area routes, i.e. summary LSAs, use the ADD OSPF RANGE or SET OSPF RANGE commands on an Area Border Router to create a range that is not advertised into other areas:

```
ADD OSPF RANGE=ipadd AREA={BACKBONE|area-number} [MASK=ipadd]
[EFFECT={ADVERTISE|DONOTADVERTISE}]

SET OSPF RANGE=ipadd [AREA={BACKBONE|area-number}]
[MASK=ipadd] [EFFECT={ADVERTISE|DONOTADVERTISE}]
```

Setting the EFFECT parameter set to DONOTADVERTISE prevents routing information for the range being advertised into other areas.

The SET IP ROUTE FILTER command has been enhanced. All filter parameters are now optional. The IP address, mask and action of the filter can be modified:

```
SET IP ROUTE FILTER=filter-id [IP=ipadd] [MASK=ipadd]
[ACTION={INCLUDE|EXCLUDE}]
[DIRECTION={RECEIVE|SEND|BOTH}] [INTERFACE=interface]
[NEXTHOP=ipadd] [POLICY=0..7]
[PROTOCOL={ANY|RIP|OSPF|EGP|STATIC|INTERFACE}]
```

## Ping Timeout

The TIMEOUT parameter in the PING and SET PING commands specifies the length of time to wait for a response to a Ping packet. The TIMEOUT parameter must now be set to a non-zero value.

## TCP Listen Ports Closed by Default

The TCP listen port 00515 is now closed by default. The CREATE LPD command opens the port. The port then remains open until the router or switch is restarted.

The X.25 DCE listen port is now closed by default. The CREATE X25C and SET X25C TCPKEEPALIVE commands open the port. The port then remains open until the router or switch is restarted.

The TCP listen port 05026 is now closed by default. The ADD STT command opens the port. The port then remains open until the router or switch is restarted.

## IPv6 Routing Enhancements

The following enhancements have been made to IPv6 routing in Software Release 2.4.1:

- IPv6 packets can be tunnelled over IPv4 interfaces
- Multiple IPv6 interfaces can use the same link-local address
- Prefixes can be manually added to Router Advertisement messages transmitted over IPv6 interfaces
- Neighbours can be manually added to the Neighbour Discovery cache
- Path MTU Discovery can be enabled, to use paths with a higher MTU than the IPv6 minimum link MTU.
- Poison reverse can be enabled on IPv6 interfaces configured for RIP.



---

*IPv6 requires a special feature licence. Contact your authorised Allied Telesyn distributor or reseller for more information.*

---

## 6-to-4 Tunneling

IPv6 packets can be tunnelled over IPv4 interfaces, as described in RFC3056 “*Connection of IPv6 Domains via IPv4 Clouds*”. The IPv4 interfaces must be correctly configured with global IPv4 addresses, and routes must be set up to ensure the router can communicate with the router at the remote end of the tunnel using IPv4.

To configure a 6-to-4 interface, first create a virtual tunnel interface, using the command:

```
ADD IPV6 6TO4 IP=ipv4add
```

where *ipv4add* is the global IPv4 address of the IPv4 interface that IPv6 packets will be transmitted and received on. This command creates a virtual tunnel interface, with an interface name of *virt*, and assigns it a link-local IPv6 address. The first tunnel created is numbered *virt0*, and each succeeding tunnel

is given the next available instance number. Each virtual interface behaves as a normal IPv6 interface. The interface is given a link-local IPv6 address with the prefix `2002::/16`, of the form:

```
2002:<ipv4-address>::<ipv4-address>
```

For example, if a PPP interface with the address 203.109.0.1 is configured as a 6-to-4 address, the IPv6 address of the `virt0` interface will be:

```
2002:cb6d:0001::cb6d:0001
```

Next, add a route to forward IPv6 packets through the tunnel, using the command:

```
ADD IPV6 ROUTE=host-ipv6add/prefix-length
      INT=6to4-tunnel-interface NEXTHOP=ipv6add
```

where *host-ipv6add* is the IPv6 address of the destination network, *ipv6add* is the link-local address of the 6-to-4 interface on the router at the remote end of the tunnel, and *6to4-tunnel-interface* is the virtual interface on the local router (e.g. `virt0`).

Tunnel interface names, instances and IPv6 addresses can be displayed for all configured tunnels, using the command:

```
SHOW IPV6 TUNNEL
```

To display information about a specific tunnel, use the command:

```
SHOW IPV6 INTERFACE=tunnel-interface
```

To remove a 6-to-4 interface, use the command:

```
DELETE IPV6 6TO4 IP=ipv4add
```

## IPv6 link-local addresses

When an IPv6 interface is created it is assigned both a global IP address and a link-local address. Additional addresses, both global and link-local, can be added to or removed from an IPv6 interface. However, the initial link-local address cannot be deleted, except by destroying the interface.

Different interfaces on an IPv6 router may have the same link-local address. Link-local addresses are those which can only be used on the local network that the interface is attached to. The link-local prefix is `fe80::/10`.

Telnetting to, and Pinging, a link-local address requires additional information because a single link-local address can belong to several interfaces, and routes are not automatically defined for link-local addresses. To Telnet to, or Ping, a link-local address, specify the global IPv6 address of the interface over which the Telnet or Ping request is to be transmitted as the source IP address, as well as the destination IPv6 address. For example, the local router has an Ethernet interface with a global IPv6 address of `3ffe:1/64` connected to the Ethernet interface on a remote router with a link-local address of `fe80::0200:cdff:fe00:a140/10`. To ping the remote router's link-local address, use the command:

```
PING SIPADDRESS=3ffe::1/64 fe80::0200:cdff:fe00:a140/10
```

## IPv6 Neighbour Discovery Enhancements

Prefixes can be included in the Router Advertisement messages transmitted over an IPv6 interface without adding the prefix to the interface's IPv6 address. These prefixes can assist nodes on the subnet with configuration of their link-local addresses.

To add a prefix to the Router Advertisement prefix list for a particular interface, use the command:

```
ADD IPV6 PREFIX=ipv6add/prefix-length INTERFACE=interface
[AUTONOMOUS={YES|NO}] [ONLINK={YES|NO}]
[PREFERRED=1..4294967295|INFINITE]
[VALID=1..4294967295|INFINITE]
```

To modify a prefix in the Router Advertisement prefix list, use the command:

```
SET IPV6 PREFIX=ipv6add/prefix-length INTERFACE=interface
[AUTONOMOUS={YES|NO}] [ONLINK={YES|NO}]
[PREFERRED=1..4294967295|INFINITE]
[VALID=1..4294967295|INFINITE]
```

To manually add a neighbour to the Neighbour Discovery cache, use the command:

```
ADD IPV6 ND=ipv6add INTERFACE=interface ETHERNET=macadd
[ISROUTER={YES|NO}]
```

To manually purge the Neighbour Discovery cache, use the command:

```
RESET IPV6 NDCACHE
```

To display information about the neighbours determined by neighbour discovery, use the command:

```
SHOW IPV6 NDCACHE
```

To display information about the neighbour discovery parameters, and the list of prefixes that are included in Router Advertisements, use the command:

```
SHOW IPV6 NDCONFIG
```

## IPv6 Path MTU Discovery

Path MTU Discovery enables the router to determine the actual maximum transmission units (MTU) of nodes on possible paths, and use a path with a higher path MTU than the IPv6 minimum link MTU, if such a path exists. Path MTU Discovery is disabled by default.

To enable Path MTU Discovery, use the command:

```
ENABLE IPV6 MTUDISCOVERY
```

To disable Path MTU Discovery, use the command:

```
DISABLE IPV6 MTUDISCOVERY
```

## IPv6 Poison Reverse on RIP

Poison reverse addresses the problem of slow convergence on RIPng routes. If one device in a network goes down, it can take a long time for the devices to recognise that routes through the crashed device are no longer available. With poison reverse, when a device goes down, the devices next to it will continue to advertise the route, but with a cost of 16. This cost indicates that the route is unavailable.

To enable poison reverse for RIP on an IPv6 interface, use the command:

```
ADD IPV6 RIP INTERFACE=interface
      [POISONREVERSE={ON|OFF|TRUE|FALSE}]
```

To display the RIP configuration for all IPv6 interfaces, use the command:

```
SHOW IPV6 RIP
```

## IGMP Proxy

Software Release 2.4.1 adds support for the host portion of RFC 2236, “*Internet Group Management Protocol, Version 2*”, to provide IGMP proxying. 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 will run a multicast routing protocol. Other routers in the tree will 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. A router may be configured with one or more downstream interfaces. An upstream interface is in the direction of the root of the tree. IGMP messages received from any downstream interfaces are proxied to the appropriate upstream interface. A router may only be configured with a single upstream interface.

The IGMP proxy will respond to IGMP query messages received via its upstream interface, and IGMP join or leave messages received via its downstream interface. It will not respond to IGMP join or leave messages received via its upstream interface, or IGMP query messages received via its downstream 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 a downstream interface receiving an IGMP join message, the router will send a IGMP join message via its upstream interface. Interfaces are added to, or removed from, multicast groups as a result of processing IGMP join or leave messages received via downstream interfaces. The IGMP proxy will only send 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, when IGMP is enabled and the router has an interface configured for IGMP proxy in the upstream direction and an interface configured for IGMP proxy in the downstream direction.

IGMP is disabled by default. To enable IGMP, use the command:

```
ENABLE IP IGMP
```

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 [DLC=1..1024]
```

To add an IP interface and specify IGMP proxying, use the command:

```
ADD IP INTERFACE=interface IPADDRESS={ipadd|DHCP}
      [IGMPPROXY={OFF|UPSTREAM|DOWNSTREAM}] [other-options...]
```

To configure IGMP proxy on an existing IP interface, use the command:

```
SET IP INTERFACE=interface
      IGMPPROXY={OFF|UPSTREAM|DOWNSTREAM}
```

The IGMPPROXY parameter specifies the status of IGMP proxying for the specified interface. If OFF is specified then the interface will not perform IGMP proxying. If UPSTREAM is specified, the interface will pass IGMP messages in the upstream direction. A router may only have one interface with the IGMP proxy direction set to UPSTREAM. If DOWNSTREAM is specified, the interface may receive IGMP messages from the downstream direction. The default is OFF.

To display information about IGMP, and multicast group membership for each IP interface, use the command:

```
SHOW IP IGMP
```

The SHOW IP IGMP command has been enhanced to display information about IGMP proxying. A new field, *IGMP Proxy*, has been added (Figure 4 on page 23 and Table 4 on page 24).

**Figure 4: Example output from the SHOW IP IGMP command.**

```
IGMP Protocol
-----
Status ..... Enabled
Default Query Interval ..... 65535 secs
Default Timeout Interval ..... 270 secs

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)
IGMP Proxy ..... Upstream
Group List .....

  No group memberships.

Interface Name ..... eth1 (DR)
IGMP Proxy ..... Downstream
Group List .....

  Group. 224.0.1.22          Last Adv. 10.194.254.254    Refresh time 184 secs
  Ports 24

  All Groups              Last Adv. 10.116.2.1      Refresh time 254 secs
  Ports 24
-----
```

**Table 4: New parameters in the output of the SHOW IP IGMP command.**

Parameter	Meaning
Last Member Query Interval	The <i>Max Response Time</i> value inserted into Group-Specific Queries sent in response to Leave Group messages, and 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.
Other Querier timeout	The length of time that remains before a multicast router decides that there is no longer another multicast router which should be the querier.
IGMP Proxy	The status of IGMP Proxy; one of "Off", "Upstream" or "Downstream".

## Setting IGMP Thresholds

The SET IP IGMP command has been enhanced to allow thresholds to be set:

```
SET IP IGMP [LMQI=1..4294967295] [LMQC=1..4294967295]
[QUERYRERESPONSEINTERVAL=1..4294967295]
[ROBUSTNESS=1..4294967295] [other-options...]
```

The LMQI parameter specifies the Last Member Query Interval (in tenths of a second) which is the Max Response Time value inserted into Group-Specific Queries sent in response to Leave Group messages. It is also the time interval 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 the ROBUSTNESS value.

The QUERYRERESPONSEINTERVAL parameter specifies the Max Response Time (in tenths of a second) inserted into the periodic General Queries. The default is 100 (10 seconds).

The ROBUSTNESS parameter specifies the Robustness Variable which 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 (0), and should not be one (1). The default is 2.

The SHOW IP IGMP command has been enhanced to display information about IGMP thresholds. Five new fields, *Last Member Query Interval*, *Last Member Query Count*, *Robustness Variable*, *Query Response Interval*, and *Other Querier timeout*, have been added (Figure 4 on page 23 and Table 4 on page 24).

## Multicast VLAN Registration (MVR)

*Multicast VLAN Registration (MVR)* is used by applications receiving multicast traffic across an Ethernet ring-based service provider network (for example, broadcasting several television channels).

MVR operates on the underlying mechanism of the *Internet Group Management Protocol (IGMP)* function, however one can be enabled or disabled without affecting the other. Receiver ports join and leave multicast streams by sending IGMP messages. If IGMP and MVR are both enabled, MVR reacts only to join and leave messages from the multicast group configured under MVR. IGMP will react to all messages.

The CPU sets up a forwarding table once MVR is configured in which each entry will match the multicast stream to an associated MAC address. The CPU then intercepts the IGMP messages and modifies the forwarding table to include or remove the receiver port as a receiver of the multicast stream. This selectively allows traffic to cross between different VLANs.

Up to five multicast VLANs can be set up on a switch, and a total of 256 group IP addresses can be added to the switch, divided up to belong to different multicast VLANs.

Whenever MVR is enabled on the switch, multicast routing must be disabled.

### Configuring MVR

MVR is disabled by default, and must be enabled on the switch to start multicast VLAN registration. To enable or disable MVR on the switch, use the commands:

```
ENABLE IP MVR
DISABLE IP MVR
```

To create MVR on, or remove MVR from the switch use the commands:

```
CREATE IP MVR VLAN=vlan-id SOURCEPORT=port-list
  RECIEVERPORT=port-list [other-options...]
DESTROY IP MVR VLAN=vlan-id
```

To add or delete an MVR IP multicast group address or range of addresses on a switch, use the command:

```
ADD IP MVR VLAN=vlan-id GROUPADDRESS=ipadd[-ipadd]
DELETE IP MVR VLAN=vlan-id GROUPADDRESS=ipadd[-ipadd]
```

To set parameters for the MVR mode, receiver ports, source ports, and ports with the immediate leave function, use the command:

```
SET IP MVR VLAN=vlan-id [IMTLEAVE=port-list] [MODE={DYNAMIC|
  COMPATIBLE}] [RECEIVERPORT=port-list]
  [SOURCEPORT=port-list]
```

To display information about the MVR configuration, use the command:

```
SHOW IP MVR [VLAN=vlan-id]
```




---

*MVR does not support tagged ports.*

---

## Enhancements to IP Multicast Switching

Rapier *i* switches support IP multicast switching used with DVMRP, PIM-SM, PIM-DM and multicast extensions to OSPF. When enabled, IP multicast packets are switched at hardware level. The software has control of how the packets are switched and where they are switched to.

The following improvements have been made to IP multicast switching:

- IP multicast switching, PIM and IGMP snooping can all operate at the same time
- VLAN tagging is fully supported within and between VLANs
- TTL in the IP header is correctly decremented.

## IPv6 Multicasting

Support for IPv6 multicasting has been added on AR300 and AR700 Series routers and the AR410 Branch Office router.

Any IPv6 host or router can send packets to an IPv6 multicast group's address. The packets will only be received by nodes that have joined that group. A node can join or leave a group at any time.



---

*To configure PIM for IPv4, use PIM commands. To configure PIM for IPv6, use PIM6 commands.*

---



---

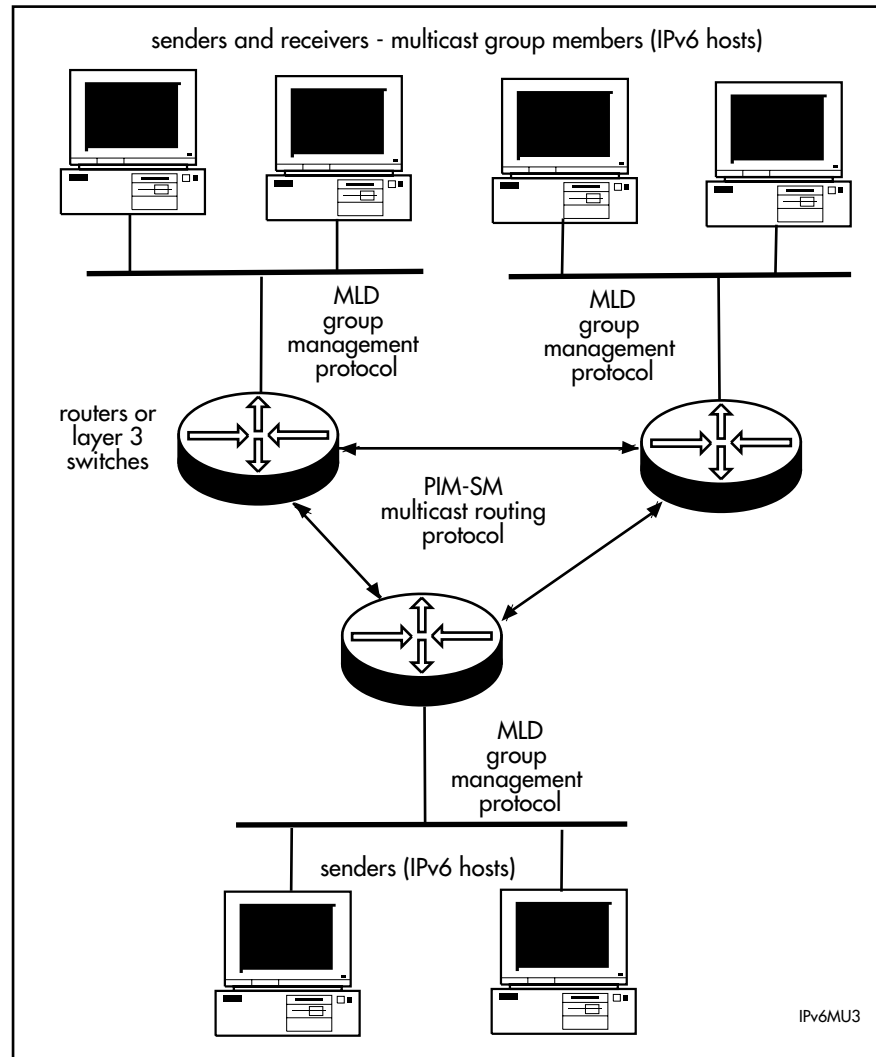
*IPv6 requires a special feature licence. Contact your authorised Allied Telesyn distributor or reseller for more information.*

---

## Overview of IPv6 Multicast Routing

For multicast routing to succeed, the router needs to know which of its interfaces are directly connected to members of each multicast group. To establish this, for IPv6, the router uses Multicast Listener Discovery for multicast group management (see "*Multicast Listener Discovery (MLD)*" on page 27). The router also needs to know which other routers to route multicast traffic to. For IPv6, the router maintains a routing table for multicast traffic with the PIM Sparse Mode (PIM-SM) multicast routing protocol (see "*Protocol Independent Multicast Sparse Mode (PIM-SM)*" on page 28). IPv6, MLD and PIM-SM for IPv6 must all be configured before the router can forward IPv6 multicast packets. The relationship between IPv6 hosts, routers and the multicasting protocols is shown in Figure 5 on page 27.

Figure 5: The IPv6 multicast environment.



## Multicast Listener Discovery (MLD)

Version 2 of the Multicast Listener Discovery Protocol (MLDv2) allows an IPv6 router to determine, for each of its directly-attached links:

- which IPv6 nodes are interested in receiving multicast traffic,
- which groups each node wishes to receive traffic from, and
- which sources a node wishes to receive traffic from, or wishes not to receive traffic from, if the node specifies a list of such sources.

MLDv2 supplies PIM Sparse Mode (PIM-SM) with information about group membership, so that PIM-SM can forward the multicast traffic appropriately to all links in the network.

Software Release 2.4.1 adds support for the router-to-router portion of MLDv2 as specified in Internet Draft "Multicast Listener Discovery Version 2 (MLDv2) for IPv6" (draft-vida-mld-v2-01). MLDv2 is interoperable with MLDv1. The router cannot act as an MLD host.

## Protocol Independent Multicast Sparse Mode (PIM-SM)

PIM Sparse Mode (PIM-SM) provides efficient communication between members of sparsely distributed groups—the types 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 routers interested in receiving traffic for a particular group receive the traffic. For IPv6, the router supports PIM Sparse Mode as specified in Internet Draft “*Protocol Independent Multicast – Sparse Mode (PIM-SM): Protocol Specification (Revised)*”, November 2001 (draft-ietf-pim-sm-v2-new-04).

### Configuring PIM-SM

PIM-SM and MLD are dependent on the router’s underlying unicast routing protocols. Before multicasting can function, IPv6 must be enabled and interfaces and routing must be configured. For PIM-SM multicast routing to operate on the router, interfaces must be assigned to PIM-SM, and PIM must be enabled for IPv6 on the router. Each network must also have a *bootstrap router*, and each multicast group must have at least one *Rendezvous Point* (RP) candidate.

PIM is disabled by default. To enable PIM for IPv6, use the command:

```
ENABLE PIM6
```

To disable it, use the command:

```
DISABLE PIM6
```

PIM-SM must be enabled for IPv6 on each of the interfaces over which it is to send and receive multicast routing messages and multicast packets. Each interface has a number assigned as its priority for becoming the *designated router* (DR) for its subnetwork. The higher the number, the higher the priority. If the subnetwork must choose a DR from interfaces with the same priority, or no priority, the interface with the highest IP address is chosen. The default designated router priority is 1, and this can be modified by specifying a different DRPRIORITY. Election by DR priority can be overridden for the interface by specifying ELECTBY=IPADDRESS.

To add or delete an interface for PIM-SM for IPv6, use the command:

```
ADD PIM6 INTERFACE=interface [DRPRIORITY=0..4294967295]
[ELECTBY={DRPRIORITY | IPADDRESS}]
```

```
DELETE PIM6 INTERFACE=interface
```

To modify the election mode or the designated router priority of a PIM interface for IPv6, use the command:

```
SET PIM6 INTERFACE=interface [DRPRIORITY=0..4294967295]
[ELECTBY={DRPRIORITY | IPADDRESS}]
```

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 value is 1 (lowest priority). The bootstrap router sends a *bootstrap message* to the other PIM-SM routers, containing a list of the RP candidates for

multicast groups at BSMINTERVAL seconds. To make the router a bootstrap router candidate, use the command:

```
ADD PIM6 BSRCANDIDATE [SCOPE=SITE|GLOBAL|ALL]
[PREFERENCE=0...255]
```

To modify the router's BSR candidate settings, use the command:

```
SET PIM6 BSRCANDIDATE [SCOPE=SITE|GLOBAL|ALL]
[PREFERENCE=0...255]
```

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

```
DELETE PIM6 BSRCANDIDATE
```

Each multicast group must have a *Rendezvous Point (RP)*, which is chosen from the list of available rendezvous point candidates. There must be at least one RP candidate in the PIM-SM connected network, but generally there should be several. The RP advertises itself to the current bootstrap router at an interval specified by ADINTERVAL seconds. The default ADINTERVAL 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. 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 is 192.

To configure the router as an RP candidate, use the command:

```
ADD PIM6 RPCANDIDATE GROUP=ipv6address[/prefixlength]
[PRIORITY=0...255]
```




---

*The router's PRIORITY is the same for all multicast groups of the same scope, for which it is a rendezvous point candidate.*

---

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

```
SET PIM6 RPCANDIDATE GROUP=ipv6address[/prefixlength]
[PRIORITY=0...255]
```

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

```
DELETE PIM6 RPCANDIDATE
```

To display information about PIM IPv6 configuration, routes, neighbours or debugging, use the command:

```
ENABLE PIM6
DEBUG={ALL|ASSERT|BSR|C-RP-ADV|HELLO|JOIN|REGISTER} [, ...]
```

To restart all PIM IPv6 processes on an interface, which resets the PIM timers, route information and counters for the interface, use the command:

```
RESET PIM6 INTERFACE=interface
```

Timers for PIM IPv6 operations have default values that will suit most networks, and should not generally be modified. If they need to be modified, use the command:

```
SET PIM6 [ADVINTERVAL={10...15000|DEFAULT}]
        [BSMINTERVAL={10...15000|DEFAULT}]
        [KEEPALIVETIME={10...65535|DEFAULT}]
        [JPINTERVAL={1...65535|DEFAULT}]
        [PROBETIME={1...65535|DEFAULT}]
        [SUPPRESSIONTIME={1...65535|DEFAULT}]
```




---

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

---

## Configuring MLD

To enable an interface to listen for and respond to MLD traffic, IPv6 must be enabled and configured on that interface. Then enable MLD, using the command:

```
ENABLE IPV6 MLD
```

and enable MLD on the desired interface, using the command:

```
ENABLE IPV6 MLD INTERFACE=interface
```

If MLD is enabled on an interface before it is globally enabled, the interface will begin processing traffic as soon as MLD is enabled.




---

*MLD must be enabled on all interfaces on which PIM for IPv6 is enabled.*

---

MLD can be globally disabled, using the command:

```
DISABLE IPV6 MLD
```

or for a single interface, using the command:

```
DISABLE IPV6 MLD INTERFACE=interface
```

## OSPF/RIP Static Export Control Parameter

RIP and OSPF offer mechanisms for maintaining and distributing dynamic routing tables. In addition to these dynamic routes, it is also possible to define static routes, typically covering routes not provided by the dynamic protocols.

The implementation of RIP and OSPF has been enhanced to allow a user to specify whether or not static routing configurations should be propagated.

When configuring RIP or OSPF on an interface, new `STATICEXPORT` parameter specifies whether or not static routing information will be propagated from an interface. If `YES` is specified, static routes are included in routing exports. If `NO` is specified, static routes are omitted from routing exports. The default is `YES`.

To add a RIP neighbour and specify whether or not static routing is propagated from an interface, use the command:

```
ADD IP RIP INTERFACE=interface [STATICEXPORT={YES|NO}]
    [other-options...]
```

To modify the attributes of the RIP neighbour with static routing either enabled or disabled, use the command:

```
SET IP RIP INTERFACE=interface [STATICEXPORT={YES|NO}]
    [other-options...]
```

The `SHOW IP RIP` command has been enhanced to display information about static routing. A new field, *Static*, has been added (Figure 6 on page 31 and Table 5 on page 31).

**Figure 6: Example output from the SHOW IP RIP command.**

Interface	IP Address	Send	Receive	Demand	Static	Auth	Password
eth0	-	COMP	BOTH	NO	YES	NO	
ppp0	172.16.249.34	RIP1	RIP2	YES	NO	PASS	*****
ppp1	172.16.250.2	RIP2	NONE	YES	YES	PASS	NOT SET

**Table 5: New parameters in the output of the SHOW IP RIP command.**

Parameter	Meaning
Static	Whether or not static routes are exported; one of "YES" or "NO".

To configure OSPF routing with static routing either enabled or disabled, use the command:

```
SET OSPF [STATICEXPORT={YES|NO}] [other-options...]
```

The `SHOW OSPF` command has been enhanced to display information about static routing. A new field, *Export static routes*, has been added (Figure 7 on page 32 and Table 6 on page 32).

**Figure 7: Example output from the SHOW OSPF command.**

```

Router ID ..... 123.234.143.231
OSPF module status ..... Enabled
Area border router status ..... Yes
AS border router status ..... Disabled
PTP stub network generation ..... Enabled
External LSA count ..... 10234
External LSA sum of checksums ... 1002345623
New LSAs originated ..... 10345
New LSAs received ..... 34500
RIP ..... Off
Export static routes ..... Yes
Dynamic interface support ..... None
Number of active areas ..... 10
Logging ..... Disabled
Debugging ..... Disabled
AS external default route:
  Status ..... Disabled
  Type ..... 1
  Metric ..... 1

```

**Table 6: New parameters displayed in the output of the SHOW OSPF command .**

Parameter	Meaning
Export static routes	Whether or not static routing information is exported to the OSPF routing domain; one of "Yes" or "No".

## Encryption Key Support for IPv6

IPv6 addresses can be specified in ENCO keys using the command:

```

CREATE ENCO KEY=key-id
  TYPE={DES|3DES2KEY|3DESINNER|3DESOUTER|GENERAL|RSA}
  IPADDRESS=ipv6add [DESCRIPTION=description]
  [FILE=filename] [FORMAT={HEX|NIQ|SSH}] [LENGTH=length]
  [MODULE=module-id] [{RANDOM|VALUE=value}]

```

If an IPv6 address is specified with the IPADDRESS parameter, that address is associated with the key. The ISAKMP and SSH modules use the address as an identifier to find the RSA public key of a remote peer.

The address can be changed for an existing key using the command:

```

SET ENCO KEY=key-id [DESCRIPTION=description]
  [IPADDRESS={ipv4add|ipv6add}] [MODULE=module-id]

```

The IPv6 address associated with a particular key can be displayed using the command:

```

SHOW ENCO KEY=key-id

```

## AppleTalk Filtering

AppleTalk filters provide a mechanism for determining whether or not to process packets received over network interfaces. There are three types of AppleTalk filters:

- Datagram Delivery Protocol (DDP) filters
- Routing Update (RMTP) filters
- Zone (ZIP) filters

A filter is a collection of filter entries, where each filter entry specifies a pattern to match and an action to execute upon a successful match. No two filter entries with the same pattern are allowed in a filter. When a packet matches one of the patterns in the filter, the filter then determines what action to take with the packet.

For each of the three types of AppleTalk filters, up to 100 filters numbered 0 to 99 can be defined.

Within a filter, each filter entry is identified by an entry number, from 1 to 65535, and filter entries are ordered according to entry number (in increasing order). A filter entry with a low entry number has precedence over a filter entry with a high entry number. For example, a packet will be matched against *filter entry1* before being matched against *filter entry2*. A new filter entry can be inserted between existing entries and the entry numbers will adjust accordingly.



---

*A filter may be associated with one or more interfaces. However, an interface may only have one type of filter associated with it.*

---

When an interface receives, or is about to transmit a packet, the filter specific to the packet received or about to be transmitted is inspected to see if that packet is allowed to be received or transmitted on that interface.

The search process within the filter is done from the first filter entry until a filter entry with a matching pattern is found or no more filter entries are available. If a matching filter entry is found, action taken by the interface will follow the action specified by the filter entry. Otherwise, the filters default action is implemented.

## DDP Packet Filtering

The Datagram Delivery Protocol (DDP) provides a process-to-process, best-effort delivery service across an internet. DDP or packet filters are configured to match packets against the following filter entries:

- source network number or source network range
- destination network number or destination network range
- source node number or node number range
- destination node number or node number range
- source socket number or source socket number range
- destination socket number or destination socket number range
- direction of the packet, i.e. one of incoming or outgoing

Each filter entry has an action to execute upon a successful match, either ALLOW or DENY.

If an interface associated with a packet filter receives a packet that does not match any filter entry on the filter, then by default the packet is discarded. This means that in effect every DDP packet filter has an implicit ‘discard all’ at the end of the entry list.

The application of DDP packet filtering enables the logging and counting of packets allowed or denied by a filter. This allows a network manager to closely monitor packets received by the router.

To define a DDP packet filter, or add another entry to a DDP packet filter, use the command:

```
ADD APPLE PACKETFILTER=filter-id [ENTRY=entry]
[SNETWORK={network-range | ANY}]
[DNETWORK={network-range | ANY}] [SNODE={node-range | ANY}]
[DNODE={node-range | ANY}] [SSOCKET={socket-range | ANY}]
[DSOCKET={socket-range | ANY}] [DIRECTION={IN | OUT}]
[LOG={YES | NO | ON | OFF | TRUE | FALSE}] ACTION={DENY | ALLOW}
```

To modify an entry in a DDP packet filter use the command:

```
SET APPLE PACKETFILTER=filter-id ENTRY=entry
[SNETWORK={network-range | ANY}]
[DNETWORK={network-range | ANY}] [SNODE={node-range | ANY}]
[DNODE={node-range | ANY}] [SSOCKET={socket-range | ANY}]
[DSOCKET={socket-range | ANY}] [DIRECTION={IN | OUT}]
[LOG={YES | NO | ON | OFF}] [ACTION={DENY | ALLOW}]
```

To delete an entry from a DDP packet filter, or delete an entire DDP packet filter, use the command:

```
DELETE APPLE PACKETFILTER=filter-id [ENTRY=entry]
```

To display information about DDP packet filters, use the command:

```
SHOW APPLE PACKETFILTER [=filter-id]
```

Once a DDP packet filter has been defined, it must be associated with an AppleTalk interface (port) using the command:

```
ADD APPLE PORT INTERFACE=interface
[PACKETFILTER={filter-id | NONE}] [other-options...]
```

To modify or delete the DDP packet filter associated with an AppleTalk interface (port), use the command:

```
SET APPLE PORT=port [PACKETFILTER={filter-id | NONE}]
[other-options...]
```

## RTMP or Routing Update Filtering

The Routing Table Maintenance Protocol (RTMP) manages routing information for AppleTalk networks. RTMP or route filters determine which specific route information is accepted and/or advertised by the RTMP process on a router.

A RTMP packet filter entry is characterised by:

- network number or network number range to match
- direction, one of “send” or “receive”
- action, one of “allow” or “deny”

A RTMP filter is associated with one or more interfaces.

Each time a routing update packet is received on an interface each route listed in the packet is checked against the filter entries associated with that interface. The packet is then either dropped or processed further by the routing table. Note that a route entry in a routing update packet that is exactly the same as a filter entry, or is within the network range of a filter entry, is defined as a match. If a route entry in a routing update packet is beyond the network range of a filter entry this is not a match.

Each time a routing update packet is transmitted to an interface each routing entry advertised by the RTMP process is checked against filter entries associated with that interface before routes are transmitted in a routing update packet.

This filtering mechanism also applies to RTMP reply packets generated by the router as the result of a routing query from either a host and/or router.

If route does not match any entries in the RTMP packet filter then by default the route processed as normal (i.e. sent or received).

To define an RTMP route filter, or add another entry to an RTMP route filter, use the command:

```
ADD APPLE ROUTEFILTER=filter-id [ENTRY=entry]
    NETWORK=network-range [DIRECTION={IN|OUT}]
    [LOG={YES|NO|ON|OFF}] ACTION={DENY|ALLOW}
```

To modify an entry in an RTMP route filter, use the command:

```
SET APPLE ROUTEFILTER=filter-id ENTRY=entry
    [NETWORK=network-range] [DIRECTION={IN|OUT}]
    [LOG={YES|NO|ON|OFF}] [ACTION={DENY|ALLOW}]
```

To delete an entry from an RTMP route filter, or to delete an entire RTMP route filter, use the command:

```
DELETE APPLE ZONEFILTER=filter-id [ENTRY=entry]
```

To display information about RTMP packet filters, use the command:

```
SHOW APPLE ROUTEFILTER[=filter-id]
```

Once an RTMP route filter has been defined, it must be associated with an AppleTalk interface (port) using the command:

```
ADD APPLE PORT INTERFACE=interface
    [ROUTEFILTER={filter-id|NONE}] [other-options...]
```

To modify or delete the RTMP route filter associated with an AppleTalk interface (port), use the command:

```
SET APPLE PORT=port [ROUTEFILTER={filter-id|NONE}]
    [other-options...]
```

## Zone Filtering

The Zone Information Protocol (ZIP) manages the relationship between network numbers and zone names. ZIP or zone filters prevent hosts accessing higher layer services on other non-routers in other zones.

Note that ZIP filtering does not affect the exchange of zone information between routers, only the exchange of zone information between routers and hosts. Filtering the exchange of zone information between routers would result in the abnormal operation of routers in the network. To prevent a router from

accessing a particular zone, implement RTMP filtering for the network associated with that zone.

A ZIP filter entry is characterised by:

- zone name to match
- action, one of “allow” or “deny”

A ZIP filter is associated with an interface. All replies to zone name requests coming from that interface are checked against the filter to see if the zone(s) in the reply are allowed to be advertised or not. If a match with a filter entry is made then the request is either allowed or denied. If no matching filter entry is found, then the zone is advertised in a reply packet.

To define a ZIP packet filter, or add another entry to a ZIP packet filter, use the command:

```
ADD APPLE ZONEFILTER=filter-id [ENTRY=entry] ZONE=zone-name
[LOG={YES|NO|ON|OFF}] ACTION={DENY|ALLOW}
```

To modify an entry in a ZIP packet filter, use the command:

```
SET APPLE ZONEFILTER=filter-id ENTRY=entry [ZONE=zone-name]
[LOG={YES|NO|ON|OFF}] [ACTION={DENY|ALLOW}]
```

To delete an entry from a ZIP packet filter, or to delete an entire ZIP packet filter, use the command:

```
DELETE APPLE ZONEFILTER=filter-id [ENTRY=entry]
```

To display information about ZIP packet filters, use the command:

```
SHOW APPLE ZONEFILTER[=filter-id]
```

Once a ZIP packet filter has been defined, it must be associated with an AppleTalk interface (port) using the command:

```
ADD APPLE PORT INTERFACE=interface
[ZONEFILTER={filter-id|NONE}] [other-options...]
```

To modify or delete the ZIP packet filter associated with an AppleTalk interface (port), use the command:

```
SET APPLE PORT=port [ZONEFILTER={filter-id|NONE}]
[other-options...]
```

## DHCP Extended Client ID

Software Release 2.4.1 adds support for the extended Client ID to both the DHCP client and server. DHCP servers use the Client ID field in DHCP requests to identify remote clients. An extended client ID is used when connecting multiple router or switch interfaces to the same DHCP server.

A new command:

```
SET DHCP EXTENDID={ON|OFF}
```

configures the use of extended Client IDs. The EXTENDID parameter specifies whether DHCP clients will use an extended client ID when communicating with a DHCP server. If OFF is specified, the client ID value is the hardware

address of the client interface. If ON is specified, the client ID value is extended to include an internal interface identifier, uniquely distinguishing different interfaces on a device. The default is OFF.

```
SHOW DHCP
```

The SHOW DHCP command has been significantly enhanced to display information about the use of the extended Client ID and the DHCP client process (Figure 8 on page 37, Table 7 on page 37).

**Figure 8: Example output from the SHOW DHCP command.**

```
DHCP Server

State ..... enabled
BOOTP Status ..... enabled
DEBUG Status ..... enabled
Extended Client ID ..... enabled
Policies ..... poll
                prnt
Ranges ..... develop ( 202.36.163.6 - 202.36.163.22 )
                remote ( 192.168.100.92 - 192.168.100.124 )
In Messages ..... 3
Out Messages ..... 3
In DHCP Messages ..... 3
Out DHCP Messages ..... 3
In BOOTP Messages ..... 0
Out BOOTP Messages ..... 0

DHCP Client

Interface ..... eth0
Client Identifier ..... 00-00-cd-03-b3-4c-00-80-00-01
State ..... bound
Server ..... 10.194.0.10
Assigned Domain .....
Assigned IP ..... 10.194.0.1
Assigned Mask ..... 255.255.255.255
Assigned Gateway ..... 0.0.0.0
Assigned DNS ..... 0.0.0.0
```

**Table 7: Parameters displayed in the output of the SHOW DHCP command.**

Parameter	Meaning
State	The status of the DHCP server; one of "enabled" or "disabled".
BOOTP Status	The status of BOOTP serving; one of "enabled" or "disabled".
Extended Client ID	Whether extended client IDs are transmitted by this device; one of "enabled" or "disabled".
BOOTP Status	The status of BOOTP serving; one of "enabled" or "disabled".
Policies	A list of the policies that have been defined.
Ranges	A list of the ranges that have been defined.
In Messages	The total number of DHCP or BOOTP messages received.
Out Messages	The total number of DHCP or BOOTP messages transmitted.
In DHCP Messages	The number of DHCP messages received.
Out DHCP Messages	The number of DHCP messages transmitted.
In BOOTP Messages	The number of BOOTP messages received.

**Table 7: Parameters displayed in the output of the SHOW DHCP command.**

Parameter	Meaning
Out BOOTP Messages	The number of BOOTP messages transmitted.
Interface	The interface(s) this client is active on.
Client Identifier	The identifying token used in DHCP messages for this client.
State	The current state of the DHCP client; one of Renewing, Rebinding, Selecting, Requesting, Bound, Init.
Server	The DHCP server this client is connected to.
Assigned Domain	The domain name provided for this client by the DHCP server.
Assigned IP	The IP address assigned to this client by the DHCP server.
Assigned Mask	The IP address mask matching the address assigned to this client.
Assigned Gateway	The network gateway IP address provided by the DHCP server.
Assigned DNS	The Domain Name Server IP address provided by the DHCP server.

## IPsec Support for IPv6

IPsec is mandatory in IPv6. In conjunction with encryption key handling (see “*Encryption Key Support for IPv6*” on page 32), it provides authentication and encryption to all IPv6 traffic.

IPsec policies can be created for IPv6 traffic using the command:

```
CREATE IPSEC POLICY=name INTERFACE=interface
ACTION={DENY|IPSEC|PERMIT} IPVERSION={4|6}
[BUNDLESPECIFICATION=bundlespecification-id]
[GROUP={0|1|2}] [ICMPTYPE={list|NDALL}]
[ISAKMPPOLICY=isakmp-policy-name]
[KEYMANAGEMENT={ISAKMP|MANUAL}]
[LADDRESS={ANY|ipv6add[/prefix-length] | ipv6add-ipv6add}]
[LNAME={ANY|system-name}] [LPORT={ANY|OPAQUE|port}]
[PEERADDRESS={ipv6add|ANY|DYNAMIC}] [POSITION=pos]
[RADDRESS={ANY|ipv6add[/prefix-length] | ipv6add-ipv6add}]
[RNAME={ANY|system-name}] [RPORT={ANY|port|OPAQUE}]
[SASELECTORFROMPKT={ALL|LADDRESS|LPORT|NONE|RADDRESS|
RPORT|TRANSPORTPROTOCOL}] [TRANSPORTPROTOCOL={ANY|EGP|
ESP|GRE|ICMP|OPAQUE|OSPF|RSVP|TCP|UDP|protocol}]
[USEPFSKEY={TRUE|FALSE}]
```

Setting the IPVERSION parameter to 6 indicates that this policy applies to IPv6 traffic and that only IPv6 addresses and network connections are valid.

When both IPsec and IPv6 are configured, IPv6 neighbour discovery will only function correctly if the ICMP packet types required for neighbour discovery are forwarded by IPsec without encryption or authentication. An IPsec policy must be created with an ACTION of PERMIT and filter patterns for ICMP types 133, 134, 135, and 136 (the types that are required for IPv6 neighbour discovery). The ICMPTYPE parameter includes the NDALL option, which is equivalent to specifying all the ICMP types required for IPv6 neighbour discovery.

Existing IPsec policies can be modified using the command:

```
SET IPSEC POLICY=name [other-options...]
```

ISAKMP policies can be created for key management for IPv6 traffic flows using the command:

```
CREATE ISAKMP POLICY=name PEER={ ipv6add|ANY } IPVERSION={ 4|6 }
  [AUTHTYPE={ PRESHARED|RSAENCR|RSASIG }]
  [DHEXONENTLENGTH=160..1023]
  [ENCALG={ 3DES2KEY|3DESINNER|3DESOUTER|DES }]
  [EXPIRYKBYTES=1..1000] [EXPIRYSECONDS=600..31449600]
  [GROUP={ 0|1|2 }] [HASHALG={ SHA|MD5 }] [KEY=0..65535]
  [LOCALID={ ipv6add|domainname|user-domainname|dist-name }]
  [LOCALRSAKEY=0..65535] [MODE={ MAIN|AGGRESSIVE }]
  [MSGRETRYLIMIT=0..1024] [MSGTIMEOUT=1..86400]
  [PHASE2XCHGLIMIT={ NONE|1..1024 }]
  [POLICYFILENAME=filename]
  [PRENEGOTIATE={ ON|OFF|TRUE|FALSE }]
  [SENDDLETES={ ON|OFF|TRUE|FALSE }]
  [SENDNOTIFY={ ON|OFF|TRUE|FALSE }]
  [SENDIDALWAYS={ ON|OFF|TRUE|FALSE }]
  [SETCOMMITBIT={ ON|OFF|TRUE|FALSE }]
  [REMOTEID={ ipv6add|domainname|user-domainname|dist-name }]
```

Connections can be accepted from any IPv6 address by setting the PEER parameter to ANY.

Existing ISAKMP policies can be modified using the command:

```
SET ISAKMP POLICY=name [other-options...]
```

## 3DES Outer CBC Encryption Mode

IPsec Security Association (SA) specifications can now include 168-bit 3DES Outer CBC Encryption Mode. Encryption keys can also be created for use with this mode. This mode can be specified by the option 3DESOUTER in the relevant parameter in the following commands:

```
CREATE ENCO KEY=key-id
  TYPE={ DES|3DES2KEY|3DESINNER|3DESOUTER|GENERAL|RSA }
  [other-options...]

CREATE IPSEC SASPECIFICATION=spec-id
  KEYMANAGEMENT={ ISAKMP|MANUAL } PROTOCOL={ AH|COMP|ESP }
  [ENCALG={ 3DES2KEY|3DESOUTER|3DESINNER|DES|NULL }]
  [other-options...]

CREATE ISAKMP POLICY=name PEER={ ipv4add|ipv6add|ANY }
  [ENCALG={ 3DES2KEY|3DESINNER|3DESOUTER|DES }]
  [other-options...]

SET IPSEC SASPECIFICATION=spec-id
  [ENCALG={ 3DES2KEY|3DESINNER|3DESOUTER|DES|NULL }]
  [other-options...]

SET ISAKMP POLICY=name
  [ENCALG={ 3DES2KEY|3DESINNER|3DESOUTER|DES }]
  [other-options...]
```



*3DES encryption is subject to government export controls, and requires a special feature licence. Contact your authorised Allied Telesyn distributor or reseller for more information.*

## BGP-4 Triggers

The Trigger Facility can be used to automatically run specified command scripts when particular triggers are activated. When a trigger is activated by an event, parameters specific to the event are passed to the script that is run. For a full description of the Trigger Facility, see the *Trigger Facility* chapter in the *Software Reference*.



*BGP-4 requires a special feature licence. Contact your authorised Allied Telesyn distributor or reseller for more information.*

Triggers can be created for two BGP events: when the router runs low on memory, and when a peer changes state.

<b>Event</b>	MEMORY
<b>Description</b>	The MEMORY event activates a trigger when BGP has had to start dropping routes because of low memory.
<b>Parameters</b>	There are no command parameters for this event.
<b>Script Arguments</b>	There are no arguments to pass to the script.

<b>Event</b>	PEERSTATE
<b>Description</b>	The PEERSTATE event activates a trigger whenever a state change occurs that matches the peer, state and direction conditions. If the state is ANY and the direction is BOTH, two triggers will be generated, one for leaving the old state and one for entering the new state.
<b>Parameters</b>	The following command parameter(s) can be specified in the CREATE/SET TRIGGER commands:

Parameter	Description
PEER={ANY <ipaddress>}	The IP address of the peer. The trigger is only activated for state changes from this peer. This parameter is required in the CREATE TRIGGER command for BGP triggers, and is optional in the SET TRIGGER command, unless the trigger event is changing from MEMORY to PEERSTATE.
STATE={IDLE CONNECT ACTIVE OPENSENT OPENCONFIRM ESTABLISHED ANY}	The state for which the trigger is activated. This parameter is required in the CREATE TRIGGER command for BGP triggers, and is optional in the SET TRIGGER command, unless the trigger event is changing from MEMORY to PEERSTATE.
DIRECTION={ENTER LEAVE BOTH}	The direction of the state change for which the trigger is activated. This parameter is required in the CREATE TRIGGER command for BGP triggers, and is optional in the SET TRIGGER command, unless the trigger event is changing from MEMORY to PEERSTATE.

**Script Arguments** The trigger passes the following argument(s) to the script:

Argument	Description
%1	The peer ID of the peer which has just undergone the state change.
%2	The state just left or entered.

---

Argument	Description
%3	Whether the state was left or entered.

---

**Example** To create trigger 1 that activates whenever BGP is low on memory, initiating the script MEMLOW.SCP, use the command:

```
CREATE TRIGGER=1 MODULE=BGP EVENT=MEMORY SCRIPT=MEMLOW.SCP
REPEAT=YES
```

To create trigger 2 to activate whenever the BGP peer with IP address 172.30.1.2 leaves the ESTABLISHED state, initiating the script PEERDOWN.SCP, use the command:

```
CREATE TRIGGER=2 MODULE=BGP EVENT=PEERSTATE PEER=172.30.1.2
STATE=ESTABLISHED DIRECTION=LEAVE SCRIPT=PEERDOWN.SCP
REPEAT=YES
```

To modify trigger 2 to activate when any BGP peer leaves the ESTABLISHED state, use the command:

```
SET TRIGGER=2 PEER=ANY
```

---

## Availability

---

Software Release 2.4.1 is available immediately as a FLASH release for upgrading existing routers, switches and switching routers. The release file can be downloaded directly from the Software Updates area of the Allied Telesyn web site at [www.alliedtelesyn.co.nz/support/updates/patches.html](http://www.alliedtelesyn.co.nz/support/updates/patches.html) or from the support site for your switch: [www.alliedtelesyn.co.nz/support/rapier/](http://www.alliedtelesyn.co.nz/support/rapier/) or [www.alliedtelesyn.co.nz/support/ar800/](http://www.alliedtelesyn.co.nz/support/ar800/).

Software releases must be licenced and require a password to activate. To obtain a licence and password, contact your authorised Allied Telesyn distributor or reseller.

---

## Installation

---

There are no known issues upgrading from Software Releases 2.3.1 or 2.3.2 to Software Release 2.4.1.

If you are downloading the release to a switch model with a Graphical User Interface, make sure you download the correct resource file, as well as the Software Release file.