Chapter 3
Configuring Basic IPv6 Connectivity

This chapter explains how to get a ProCurve Routing Switch that supports IPv6 up and running. To configure basic IPv6 connectivity, you must do the following:

- Enable IPv6 routing globally on the ProCurve Routing Switch.
- Configure an IPv6 address or explicitly enable IPv6 on each router interface over which you plan to forward IPv6 traffic.
- Configure IPv4 and IPv6 protocol stacks. (This step is mandatory only if you want a router interface to send and receive both IPv4 and IPv6 traffic.)

The following configuration tasks are optional:

- Configure IPv6 Domain Name Server (DNS) resolver
- Configure ECMP Load Sharing for IPv6
- Configure IPv6 ICMP.
- Configure the IPv6 neighbor discovery feature.
- Change the IPv6 MTU.
- Configure an unnumbered interface.
- Configure static neighbor entries.
- Limit the hop count of an IPv6 packet.
- Configure Quality of Service (QoS) for IPv6 traffic

Enabling IPv6 Routing

By default, IPv6 routing is disabled. To enable the forwarding of IPv6 traffic globally on the Routing Switch, enter the following command:

```
ProCurveRS(config)# ipv6 unicast-routing
```

Syntax: [no] ipv6 unicast-routing

To disable the forwarding of IPv6 traffic globally on the HP device, enter the **no** form of this command.

Configuring IPv6 on Each Router Interface

To forward IPv6 traffic on a router interface, the interface must have an IPv6 address, or IPv6 must be explicitly enabled. By default, an IPv6 address is not configured on a router interface.

If you choose to configure a global or site-local IPv6 address for an interface, IPv6 is also enabled on the interface. Further, when you configure a global or site-local IPv6 address, you must decide on one of the following in the low-order 64 bits:

- A manually configured interface ID.
- An automatically computed EUI-64 interface ID.

If you prefer to assign a link-local IPv6 address to the interface, you must explicitly enable IPv6 on the interface, which causes a link-local address to be automatically computed for the interface. If preferred, you can override the automatically configured link-local address with an address that you manually configure.

This section provides the following information:

- Configuring a global or site-local address with a manually configured or automatically computed interface ID for an interface.
- Automatically or manually configuring a link-local address for an interface.
- Configuring IPv6 anycast addresses

Configuring a Global or Site-Local IPv6 Address

Configuring a global or site-local IPv6 address on an interface does the following:

- Automatically configures an interface ID (a link-local address), if specified.
- Enables IPv6 on that interface.

Additionally, the configured interface automatically joins the following required multicast groups for that link:

- Solicited-node multicast group FF02:0:0:0:0:1:FF00::/104 for each unicast address assigned to the interface.
- All-nodes link-local multicast group FF02::1
- All-routers link-local multicast group FF02::2

The neighbor discovery feature sends messages to these multicast groups. For more information, see “Configuring IPv6 Neighbor Discovery” on page 3-8.

Configuring a Global or Site-Local IPv6 Address with a Manually Configured Interface ID

To configure a global or site-local IPv6 address, including a manually configured interface ID, for an interface, enter commands such as the following:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 address 2001:200:12D:1300:240:D0FF:
FE48:4672:/64
```

These commands configure the global prefix 2001:200:12d:1300::/64 and the interface ID ::240:D0FF:FE48:4672, and enable IPv6 on Ethernet interface 3/1.

Syntax: ipv6 address <ipv6-prefix>/<prefix-length>

You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

Configuring a Global or Site-Local IPv6 Address with an Automatically Computed EUI-64 Interface ID

To configure a global or site-local IPv6 address with an automatically computed EUI-64 interface ID in the low-order 64-bits, enter commands such as the following:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 address 2001:200:12D:1300::/64 eui-64
```

These commands configure the global prefix 2001:200:12d:1300::/64 and an interface ID, and enable IPv6 on Ethernet interface 3/1.

Syntax: ipv6 address <ipv6-prefix>/<prefix-length> eui-64

You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

The **eui-64** keyword configures the global or site-local address with an EUI-64 interface ID in the low-order 64 bits. The interface ID is automatically constructed in IEEE EUI-64 format using the interface's MAC address.

Configuring a Link-Local IPv6 Address

To explicitly enable IPv6 on a router interface without configuring a global or site-local address for the interface, enter commands such as the following:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 enable
```

These commands enable IPv6 on Ethernet interface 3/1 and specify that the interface is assigned an automatically computed link-local address.

Syntax: [no] ipv6 enable

NOTE: When configuring VLANs that share a common tagged interface with a Virtual Ethernet (VE) interface, HP recommends that you override the automatically computed link-local address with a manually configured unique address for the interface. If the interface uses the automatically computed address, which in the case of VE interfaces is derived from a global MAC address, all VE interfaces will have the same MAC address.

To override a link-local address that is automatically computed for an interface with a manually configured address, enter commands such as the following:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 address FE80::240:D0FF:FE48:4672 link-local
```

These commands explicitly configure the link-local address FE80::240:D0FF:FE48:4672 for Ethernet interface 3/1.

Syntax: ipv6 address <ipv6-address> link-local

You must specify the <ipv6-address> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

The **link-local** keyword indicates that the router interface should use the manually configured link-local address instead of the automatically computed link-local address.

Configuring an

Configuring IPv6 Anycast Addresses

In IPv6, an ***anycast*** address is an address for a set of interfaces belonging to different nodes. Sending a packet to an anycast address results in the delivery of the packet to the closest interface configured with the anycast address.

An anycast address looks similar to a unicast address, because it is allocated from the unicast address space. If you assign an IPv6 unicast address to multiple interfaces, it is an anycast address. On the HP device, you configure an interface assigned an anycast address to recognize the address as an anycast address.

For example, the following commands configure an anycast address on interface 2/1:

```
ProCurveRS(config)# int e 2/1
ProCurveRS(config-if-e100-2/1)# ipv6 address 2002::6/64 anycast
```

Syntax: ipv6 address <ipv6-prefix>/<prefix-length> [anycast]

IPv6 anycast addresses are described in detail in RFC 1884. See RFC 2461 for a description of how the IPv6 Neighbor Discovery mechanism handles anycast addresses.

Configuring IPv4 and IPv6 Protocol Stacks

One situation in which you must configure a Routing Switch to run both IPv4 and IPv6 protocol stacks is if it is deployed as an endpoint for an IPv6 over IPv4 tunnel. For more information, see “IPv6 Over IPv4 Tunnels” on page 8-2.

Each router interface that you want to send and receive both IPv4 and IPv6 traffic must be configured with an IPv4 address and an IPv6 address. (An alternative to configuring a router interface with an IPv6 address is to explicitly enable IPv6 using the **ipv6 enable** command. For more information about using this command, see “Configuring a Link-Local IPv6 Address” on page 3-3.)

To configure a router interface to support both the IPv4 and IPv6 protocol stacks, use commands such as the following:

```
ProCurveRS(config)# ipv6 unicast-routing
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ip address 192.168.1.1 255.255.255.0
ProCurveRS(config-if-e100-3/1)# ipv6 address 2001:200:12d:1300::/64 eui-64
```

These commands globally enable IPv6 routing on the Routing Switch and configure an IPv4 address and an IPv6 address for Ethernet interface 3/1.

Syntax: [no] ipv6 unicast-routing

To disable IPv6 traffic globally on the Routing Switch, enter the **no** form of this command.

Syntax: ip address <ip-address> <sub-net-mask> [secondary]

You must specify the <ip-address> parameter using 8-bit values in dotted decimal notation.

You can specify the <sub-net-mask> parameter in either dotted decimal notation or as a decimal value preceded by a slash mark (/).

The **secondary** keyword specifies that the configured address is a secondary IPv4 address.

To remove the IPv4 address from the interface, enter the **no** form of this command.

Syntax: ipv6 address <ipv6-prefix>/<prefix-length> [eui-64]

This syntax specifies a global or site-local IPv6 address. For information about configuring a link-local IPv6 address, see “Configuring a Link-Local IPv6 Address” on page 3-3.

You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

The **eui-64** keyword configures the global or site-local address with an EUI-64 interface ID in the low-order 64 bits. The interface ID is automatically constructed in IEEE EUI-64 format using the interface’s MAC address. If you do not specify the **eui-64** keyword, you must manually configure the 64-bit interface ID as well as the 64-bit network prefix. For more information about manually configuring an interface ID, see “Configuring a Global or Site-Local IPv6 Address” on page 3-2.

Configuring IPv6 Domain Name Server (DNS) Resolver

The Domain Name Server (DNS) resolver feature lets you use a host name to perform Telnet, ping, and traceroute commands. You can also define a DNS domain on an HP device and thereby recognize all hosts within that domain. After you define a domain name, the HP automatically appends the appropriate domain to the host and forwards it to the domain name server.

For example, if the domain “newyork.com” is defined on an HP and you want to initiate a ping to host “NYC01” on that domain, you need to reference only the host name in the command instead of the host name and its domain name. For example, you could enter either of the following commands to initiate the ping:

```
ProCurveRS# ping nyc01
ProCurveRS# ping nyc01.newyork.com
```

Defining a DNS Entry

You can define up to four DNS servers for each DNS entry. The first entry serves as the primary default address. If a query to the primary address fails to be resolved after three attempts, the next gateway address is queried (also up to three times). This process continues for each defined gateway address until the query is resolved. The order in which the default gateway addresses are polled is the same as the order in which you enter them.

Suppose you want to define the domain name of newyork.com on an HP and then define four possible default DNS gateway addresses. To do so using IPv4 addressing, you would enter the following commands:

```
ProCurveRS(config)# ip dns domain-name newyork.com
ProCurveRS(config)# ip dns server-address 209.157.22.199 205.96.7.15 208.95.7.25
201.98.7.15
```

Syntax: ip dns server-address <ip-addr> [<ip-addr>] [<ip-addr>] [<ip-addr>]

In this example, the first IP address in the **ip dns server-address...** command becomes the primary gateway address and all others are secondary addresses. Because IP address 201.98.7.15 is the last address listed, it is also the last address consulted to resolve a query.

Defining an IPv6 DNS Entry

IPv6 defines new DNS record types to resolve queries for domain names to IPv6 addresses, as well as IPv6 addresses to domain names. HP devices running IPv6 software support AAAA DNS records, which are defined in RFC 1886.

AAAA DNS records are analogous to the A DNS records used with IPv4. They store a complete IPv6 address in each record. AAAA records have a type value of 28.

To establish an IPv6 DNS entry for the device, enter the following command:

```
ProCurveRS(config)# ipv6 dns domain-name companynet.com
```

Syntax: [no] ipv6 dns domain-name <domain name>

To define an IPv6 DNS server address, enter the following command:

```
ProCurveRS(config)# ipv6 dns server-address 200::1
```

Syntax: [no] ipv6 dns server-address <ipv6-addr> [<ipv6-addr>] [<ipv6-addr>] [<ipv6-addr>]

As an example, in a configuration where ftp6.companynet.com is a server with an IPv6 protocol stack, when a user pings ftp6.companynet.com, the HP device attempts to resolve the AAAA DNS record. In addition, if the DNS server does not have an IPv6 address, as long as it is able to resolve AAAA records, it can still respond to DNS queries.

ECMP Load Sharing for IPv6

The IPv6 route table selects the best route to a given destination from among the routes in the tables maintained by the configured routing protocols (BGP4, OSPF, static, and so on). The IPv6 route table can contain more than one path to a given destination. When this occurs, the HP device selects the path with the lowest cost for insertion into the routing table. If more than one path with the lowest cost exists, all of these paths are inserted into the routing table, subject to the configured maximum number of load sharing paths (by default 4). The device uses ***Equal-Cost Multi-Path (ECMP) load sharing*** to select a path to a destination.

When the device receives traffic for a destination, and the IPv6 route table contains multiple, equal-cost paths to that destination, the device checks the ***IPv6 forwarding cache*** for a forwarding entry for the destination. The IPv6 forwarding cache provides a fast path for forwarding IPv6 traffic. The IPv6 forwarding cache contains entries that associate a destination host or network with a path (next-hop router).

If the IPv6 forwarding cache contains a forwarding entry for the destination, the HP device uses the entry to forward the traffic. If the IPv6 forwarding cache does not contain a forwarding entry for the destination, the software selects a path from among the available equal-cost paths to the destination, then creates an entry in the in the cache based on the calculation. Subsequent traffic for the same destination uses the forwarding entry. Entries remain in the IPv6 forwarding cache for one minute, then are aged out.

If the path selected by the device becomes unavailable, its entry in the IPv6 forwarding cache is removed, a new path is selected from the remaining equal-cost paths to the destination, and an entry is created in the IPv6 forwarding cache using the new path.

HP devices support the following ECMP load-sharing methods for IPv6 traffic:

- Network-based – The HP device distributes traffic across equal-cost paths based on destination network address. The software selects a path based on a calculation involving the maximum number of load-sharing paths allowed and the actual number of paths to the destination network. This is the default ECMP load-sharing method for IPv6.
- Host-based – The HP device uses a simple round-robin mechanism to distribute traffic across the equal-cost paths based on destination host IP address. The device uses this ECMP load-sharing method for IPv6 if you explicitly configure it to do so.

You can manually disable or enable ECMP load sharing for IPv6, specify the number of equal-cost paths the device can distribute traffic across, and configure the device to use the host-based ECMP load-sharing method instead of the network-based method. In addition, you can display information about the status of ECMP load-sharing on the device, as well as the entries in the IPv6 forwarding cache.

Disabling or Re-Enabling ECMP Load Sharing for IPv6

ECMP load sharing for IPv6 is enabled by default. To disable the feature, enter the following command:

```
ProCurveRS(config)# no ipv6 load-sharing
```

If you want to re-enable the feature after disabling it, enter the following command:

```
ProCurveRS(config)# ipv6 load-sharing
```

Syntax: [no] ipv6 load-sharing

Changing the Maximum Number of Load Sharing Paths for IPv6

By default, IPv6 ECMP load sharing allows traffic to be balanced across up to four equal paths. You can change the maximum number of paths the device supports to a value from 2 – 8.

To change the number of ECMP load sharing paths for IPv6, enter a command such as the following:

```
ProCurveRS(config)# ipv6 load-sharing 8
```

Syntax: [no] ipv6 load-sharing [<num>]

The <num> parameter specifies the number of paths and can be from 2 – 8. The default is 4.

Changing the ECMP Load-Sharing Method for IPv6

HP devices can perform ECMP load-sharing for IPv6 traffic based on destination host address or destination network. The default is network-based IP load sharing. If you want to enable the device to perform host-based IP load sharing instead, enter the following command:

```
ProCurveRS(config)# ipv6 load-sharing by-host
```

Syntax: [no] ipv6 load-sharing by-host

This command enables host-based ECMP load sharing on the device. The command also disables network-based ECMP load-sharing at the same time.

Displaying ECMP Load-Sharing Information for IPv6

To display the status of ECMP load sharing for IPv6, enter the following command:

```
ProCurveRS# show ipv6
Global Settings

  unicast-routing enabled, hop-limit 64
  No Inbound Access List Set
  No Outbound Access List Set
  Prefix-based IPv6 Load-sharing is Enabled, Number of load share paths: 4
```

Syntax: show ipv6

You can display the entries in the IPv6 forwarding cache; for example

:

```
ProCurveRS# show ipv6 cache
Total number of cache entries: 10
     IPv6 Address                          Next Hop                        Port
1    5000:2::2                             LOCAL                           tunnel 2
2    2000:4::106                           LOCAL                           ethe 2
3    2000:4::110                           DIRECT                          ethe 2
4    2002:c0a8:46a::1                      LOCAL                           ethe 2
5    fe80::2e0:52ff:fe99:9737              LOCAL                           ethe 2
6    fe80::ffff:ffff:feff:ffff             LOCAL                           loopback 2
7    fe80::c0a8:46a                        LOCAL                           tunnel 2
8    fe80::c0a8:46a                        LOCAL                           tunnel 6
9    2999::1                               LOCAL                           loopback 2
10   fe80::2e0:52ff:fe99:9700              LOCAL                           ethe 1
```

Syntax: show ipv6 cache [<index-number> | <ipv6-prefix>/<prefix-length> | <ipv6-address> | ethernet <port> | ve <number> | tunnel <number>]

Configuring IPv6 ICMP

As with the Internet Control Message Protocol (ICMP) for IPv4, ICMP for IPv6 provides error and informational messages. HP's implementation of the stateless autoconfiguration, neighbor discovery, and path MTU discovery features use ICMP messages.

This section explains how to configure the following IPv6 ICMP features:

- ICMP rate limiting.
- ICMP redirects.

Configuring ICMP Rate Limiting

You can limit the rate at which IPv6 ICMP error messages are sent out on a network. IPv6 ICMP implements a token bucket algorithm.

To illustrate how this algorithm works, imagine a virtual bucket that contains a number of tokens. Each token represents the ability to send one ICMP error message. Tokens are placed in the bucket at a specified interval until the maximum number of tokens allowed in the bucket is reached. For each error message that ICMP sends, a token is removed from the bucket. If ICMP generates a series of error messages, messages can be sent until the bucket is empty. If the bucket is empty of tokens, error messages cannot be sent until a new token is placed in the bucket.

You can adjust the following elements related to the token bucket algorithm:

- The interval at which tokens are added to the bucket. The default is 100 milliseconds.
- The maximum number of tokens in the bucket. The default is 10 tokens.

For example, to adjust the interval to 1000 milliseconds and the number of tokens to 100 tokens, enter the following command:

```
ProCurveRS(config)# ipv6 icmp error-interval 1000 100
```

Syntax: ipv6 icmp error-interval <interval> [<number-of-tokens>]

The interval in milliseconds at which tokens are placed in the bucket can range from 0 – 2147483647. The maximum number of tokens stored in the bucket can range from 1 – 200.

NOTE: If you retain the default interval value or explicitly set the value to 100 milliseconds, output from the **show run** command does not include the setting of the **ipv6 icmp error-interval** command because the setting is the default.

Also, if you configure the interval value to a number that does not evenly divide into 100000 (100 milliseconds), the system rounds up the value to a next higher value that does divide evenly into 100000. For example, if you specify an interval value of 150, the system rounds up the value to 200.

ICMP rate limiting is enabled by default. To disable ICMP rate limiting, set the interval to zero.

Disabling or Reenabling ICMP Redirect Messages

You can disable or re-enable the sending of ICMP redirect messages by a Routing Switch. By default, a Routing Switch can send an ICMP redirect message to a neighboring host to inform it of a better first-hop router on a path to a destination. No further configuration is required to enable the sending of ICMP redirect messages. (For more information about how ICMP redirect messages are implemented for IPv6, see “Configuring IPv6 Neighbor Discovery” on page 3-8.)

For example, to disable the sending of ICMP redirect messages on Ethernet interface 3/1, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# no ipv6 redirects
```

Syntax: [no] ipv6 redirects

To reenable the sending of ICMP redirect messages on Ethernet interface 3/1, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 redirects
```

Use the **show ipv6 interface** <interface> <port-number> command to verify that the sending of ICMP redirect messages is enabled on a particular interface.

Configuring IPv6 Neighbor Discovery

The neighbor discovery feature for IPv6 uses IPv6 ICMP messages to do the following:

- Determine the link-layer address of a neighbor on the same link.
- Verify that a neighbor is reachable.

- Track neighbor routers.

An IPv6 host is required to listen for and recognize the following addresses that identify itself:

- Link-local address.
- Assigned unicast address.
- Loopback address.
- All-nodes multicast address.
- Solicited-node multicast address.
- Multicast address to all other groups to which it belongs.

You can adjust the following IPv6 neighbor discovery features:

- Neighbor solicitation messages for duplicate address detection.
- Router advertisement messages:
    - Interval between router advertisement messages.
    - Value that indicates a router is advertised as a default router (for use by all nodes on a given link).
    - Prefixes advertised in router advertisement messages.
    - Flags for host stateful autoconfiguration.
- Amount of time during which an IPv6 node considers a remote node reachable (for use by all nodes on a given link).

Neighbor Solicitation and Advertisement Messages

Neighbor solicitation and advertisement messages enable a node to determine the link-layer address of another node (neighbor) on the same link. (This function is similar to the function provided by the Address Resolution Protocol [ARP] in IPv4.) For example, node 1 on a link wants to determine the link-layer address of node 2 on the same link. To do so, node 1, the source node, multicasts a neighbor solicitation message. The neighbor solicitation message, which has a value of 135 in the Type field of the ICMP packet header, contains the following information:

- Source address: IPv6 address of node 1 interface that sends the message.
- Destination address: solicited-node multicast address (FF02:0:0:0:0:1:FF00::/104) that corresponds the IPv6 address of node 2.
- Link-layer address of node 1.
- A query for the link-layer address of node 2.

After receiving the neighbor solicitation message from node 1, node 2 replies by sending a neighbor advertisement message, which has a value of 136 in the Type field of the ICMP packet header. The neighbor solicitation message contains the following information:

- Source address: IPv6 address of the node 2 interface that sends the message.
- Destination address: IPv6 address of node 1.
- Link-layer address of node 2.

After node 1 receives the neighbor advertisement message from node 2, nodes 1 and 2 can now exchange packets on the link.

After the link-layer address of node 2 is determined, node 1 can send neighbor solicitation messages to node 2 to verify that it is reachable. Also, nodes 1, 2, or any other node on the same link can send a neighbor advertisement message to the all-nodes multicast address (FF02::1) if there is a change in their link-layer address.

Router Advertisement and Solicitation Messages

Router advertisement and solicitation messages enable a node on a link to discover the routers on the same link.

Each configured router interface on a link sends out a router advertisement message, which has a value of 134 in the Type field of the ICMP packet header, periodically to the all-nodes link-local multicast address (FF02::1).

A configured router interface can also send a router advertisement message in response to a router solicitation message from a node on the same link. This message is sent to the unicast IPv6 address of the node that sent the router solicitation message.

At system startup, a host on a link sends a router solicitation message to the all-routers multicast address (FF01). Sending a router solicitation message, which has a value of 133 in the Type field of the ICMP packet header, enables the host to automatically configure its IPv6 address immediately instead of awaiting the next periodic router advertisement message.

Because a host at system startup typically does not have a unicast IPv6 address, the source address in the router solicitation message is usually the unspecified IPv6 address (0:0:0:0:0:0:0:0). If the host has a unicast IPv6 address, the source address is the unicast IPv6 address of the host interface sending the router solicitation message.

Entering the **ipv6 unicast-routing** command automatically enables the sending of router advertisement messages on all configured router Ethernet interfaces. You can configure several router advertisement message parameters. For information about disabling the sending of router advertisement messages and the router advertisement parameters that you can configure, see “Enabling and Disabling IPv6 Router Advertisements” on page 3-12 and “Setting IPv6 Router Advertisement Parameters” on page 3-11.

Neighbor Redirect Messages

After forwarding a packet, by default, a Routing Switch can send a neighbor redirect message to a host to inform it of a better first-hop router. The host receiving the neighbor redirect message will then readdress the packet to the better router.

A Routing Switch sends a neighbor redirect message only for unicast packets, only to the originating node, and to be processed by the node.

A neighbor redirect message has a value of 137 in the Type field of the ICMP packet header.

Setting Neighbor Solicitation Parameters for Duplicate Address Detection

Although the stateless autoconfiguration feature assigns the 64-bit interface ID portion of an IPv6 address using the MAC address of the host’s NIC, duplicate MAC addresses can occur. Therefore, the duplicate address detection feature verifies that a unicast IPv6 address is unique before it is assigned to a host interface by the stateless autoconfiguration feature. Duplicate address detection verifies that a unicast IPv6 address is unique.

If duplicate address detection identifies a duplicate unicast IPv6 address, the address is not used. If the duplicate address is the link-local address of the host interface, the interface stops processing IPv6 packets.

You can configure the following neighbor solicitation message parameters that affect duplicate address detection while it verifies that a tentative unicast IPv6 address is unique:

- The number of consecutive neighbor solicitation messages that duplicate address detection sends on an interface. By default, duplicate address detection sends three neighbor solicitation messages without any follow-up messages.
- The interval in seconds at which duplicate address detection sends a neighbor solicitation message on an interface. By default, duplicate address detection sends a neighbor solicitation message every 1 second.

NOTE: For the interval at which duplicate address detection sends a neighbor solicitation message on an interface, the HP device uses seconds as the unit of measure instead of milliseconds.

For example, to change the number of neighbor solicitation messages sent on Ethernet interface 3/1 to two and the interval between the transmission of the two messages to 9 seconds, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 nd dad attempt 2
ProCurveRS(config-if-e100-3/1)# ipv6 nd ns-interval 9
```

Syntax: [no] ipv6 nd dad attempt <number>

Syntax: [no] ipv6 nd ns-interval <number>

For the number of neighbor solicitation messages, you can specify any number of attempts. Configuring a value of 0 disables duplicate address detection processing on the specified interface. To restore the number of messages to the default value, use the **no** form of this command.

For the interval between neighbor solicitation messages, you can specify any number of seconds. HP does not recommend very short intervals in normal IPv6 operation. When a nondefault value is configured, the configured time is both advertised and used by the Routing Switch itself. To restore the default interval, use the **no** form of this command.

Setting IPv6 Router Advertisement Parameters

You can adjust the following parameters for router advertisement messages:

- The interval (in seconds) at which an interface sends router advertisement messages. By default, an interface sends a router advertisement message every 200 seconds.
- The "router lifetime" value, which is included in router advertisements sent from a particular interface. The value (in seconds) indicates if the Routing Switch is advertised as a default router on this interface. If you set the value of this parameter to 0, the Routing Switch is not advertised as a default router on an interface. If you set this parameter to a value that is not 0, the Routing Switch is advertised as a default router on this interface. By default, the router lifetime value included in router advertisement messages sent from an interface is 1800 seconds.

When adjusting these parameter settings, HP recommends that the interval between router advertisement transmission be less than or equal to the router lifetime value if the Routing Switch is advertised as a default router. For example, to adjust the interval of router advertisements to 300 seconds and the router lifetime value to 1900 seconds on Ethernet interface 3/1, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 nd ra-interval 300
ProCurveRS(config-if-e100-3/1)# ipv6 nd ra-lifetime 1900
```

Syntax: [no] ipv6 nd ra-interval <number>

Syntax: [no] ipv6 nd ra-lifetime <number>

The <number> parameter in both commands indicates any numerical value. To restore the default interval or router lifetime value, use the **no** form of the respective command.

Controlling Prefixes Advertised in IPv6 Router Advertisement Messages

By default, router advertisement messages include prefixes configured as addresses on router interfaces using the **ipv6 address** command. You can use the **ipv6 nd prefix-advertisement** command to control exactly which prefixes are included in router advertisement messages. Along with which prefixes the router advertisement messages contain, you can also specify the following parameters:

- Valid lifetime—(Mandatory) The time interval (in seconds) in which the specified prefix is advertised as valid. The default is 2592000 seconds (30 days). When the timer expires, the prefix is no longer considered to be valid.
- Preferred lifetime—(Mandatory) The time interval (in seconds) in which the specified prefix is advertised as preferred. The default is 604800 seconds (7 days). When the timer expires, the prefix is no longer considered to be preferred.
- Onlink flag—(Optional) If this flag is set, the specified prefix is assigned to the link upon which it is advertised. Nodes sending traffic to addresses that contain the specified prefix consider the destination to be reachable on the local link.
- Autoconfiguration flag—(Optional) If this flag is set, the stateless auto configuration feature can use the specified prefix in the automatic configuration of 128-bit IPv6 addresses for hosts on the local link. For more information, see “IPv6 Stateless Autoconfiguration” on page 2-4.

For example, to advertise the prefix 2001:e077:a487:7365::/64 in router advertisement messages sent out on Ethernet interface 3/1 with a valid lifetime of 1000 seconds, a preferred lifetime of 800 seconds, and the Onlink and Autoconfig flags set, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 nd prefix-advertisement 2001:e077:a487:7365::/
64 1000 800 onlink autoconfig
```

Syntax: [no] ipv6 nd prefix-advertisement <ipv6-prefix>/<prefix-length> <valid-lifetime> <preferred-lifetime> [autoconfig] [onlink]

You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

The valid lifetime and preferred lifetime is a numerical value between 0 – 4294967295 seconds. The default valid lifetime is 2592000 seconds (30 days), while the default preferred lifetime is 604800 seconds (7 days).

To remove a prefix from the router advertisement messages sent from a particular interface, use the **no** form of this command.

Setting Flags in IPv6 Router Advertisement Messages

An IPv6 router advertisement message can include the following flags:

- Managed Address Configuration—This flag indicates to hosts on a local link if they should use the stateful autoconfiguration feature to get IPv6 addresses for their interfaces. If the flag is set, the hosts use stateful autoconfiguration to get addresses as well as non-IPv6-address information. If the flag is not set, the hosts do not use stateful autoconfiguration to get addresses and if the hosts can get non-IPv6-address information from stateful autoconfiguration is determined by the setting of the Other Stateful Configuration flag.
- Other Stateful Configuration—This flag indicates to hosts on a local link if they can get non-IPv6 address autoconfiguration information. If the flag is set, the hosts can use stateful autoconfiguration to get non-IPv6-address information.

NOTE: When determining if hosts can use stateful autoconfiguration to get non-IPv6-address information, a set Managed Address Configuration flag overrides an unset Other Stateful Configuration flag. In this situation, the hosts can obtain nonaddress information. However, if the Managed Address Configuration flag is not set and the Other Stateful Configuration flag is set, then the setting of the Other Stateful Configuration flag is used.

By default, the Managed Address Configuration and Other Stateful Configuration flags are not set in router advertisement messages. For example, to set these flags in router advertisement messages sent from Ethernet interface 3/1, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 nd managed-config-flag
ProCurveRS(config-if-e100-3/1)# ipv6 nd other-config-flag
```

Syntax: [no] ipv6 nd managed-config-flag

Syntax: [no] ipv6 nd other-config-flag

To remove either flag from router advertisement messages sent on an interface, use the **no** form of the respective command.

Enabling and Disabling IPv6 Router Advertisements

If IPv6 unicast routing is enabled on an Ethernet interface, by default, this interface sends IPv6 router advertisement messages. However, by default, non-LAN interface types, for example, tunnel interfaces, do not send router advertisement messages.

To disable the sending of router advertisement messages on an Ethernet interface, enter commands such as the following:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 nd suppress-ra
```

To enable the sending of router advertisement messages on a tunnel interface, enter commands such as the following:

```
ProCurveRS(config)# interface tunnel 1
ProCurveRS(config-tnif-1)# no ipv6 nd suppress-ra
```

Syntax: [no] ipv6 nd suppress-ra

Configuring Reachable Time for Remote IPv6 Nodes

You can configure the duration (in seconds) that Routing Switch considers a remote IPv6 node reachable. By default, a router interface uses the value of 30 seconds.

The router advertisement messages sent by a router interface include the amount of time specified by the **ipv6 nd reachable-time** command so that nodes on a link use the same reachable time duration. By default, the messages include a default value of 0.

NOTE: For the interval at which a router interface sends router advertisement messages, HP uses seconds as the unit of measure instead of milliseconds.

HP does not recommend configuring a short reachable time duration, because a short duration causes the IPv6 network devices to process the information at a greater frequency.

For example, to configure the reachable time of 40 seconds for Ethernet interface 3/1, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 nd reachable-time 40
```

Syntax: [no] ipv6 nd reachable-time <seconds>

For the <seconds> parameter, you can specify any numerical value. To restore the default time, use the **no** form of this command.

Changing the IPv6 MTU

The IPv6 MTU is the maximum length of an IPv6 packet that can be transmitted on a particular interface. If an IPv6 packet is longer than an MTU, the host that originated the packet fragments the packet and transmits its contents in multiple packets that are shorter than the configured MTU. You can configure the MTU on individual interfaces. Per RFC 2460, the minimum IPv6 MTU for any interface is 1280 bytes.

For example, to configure the MTU on Ethernet interface 3/1 as 1280 bytes, enter the following commands:

```
ProCurveRS(config)# interface ethernet 3/1
ProCurveRS(config-if-e100-3/1)# ipv6 mtu 1280
```

Syntax: [no] ipv6 mtu <bytes>

You can specify between 1280 – 1500 bytes. If a nondefault value is configured for an interface, router advertisements include an MTU option.

Configuring an Unnumbered Interface

You can enable IPv6 on a tunnel interface but not assign an IPv6 address to the interface. The unnumbered interface feature is useful when you are connecting two isolated IPv6 domains over an IPv4 infrastructure. In this situation, an IPv6 address on the tunnel interface might not serve a purpose.

For example, to configure tunnel interface 1 as an unnumbered port and specify the global IPv6 address of Ethernet 3/1 as the source address, enter the following commands:

```
ProCurveRS(config)# interface tunnel 1
ProCurveRS(config-tnif-1)# ipv6 unnumbered ethernet 3/1
```

Syntax: ipv6 unnumbered <interface> <number>

The syntax of the **ipv6 unnumbered** command requires that you specify the interface type and number of a physical port. The software uses the global IPv6 address of the specified physical port as the source address for IPv6 packets generated by the unnumbered interface.

IPv6 packets that are originated from an unnumbered interface use the global IPv6 address of the interface specified in the **ipv6 unnumbered** command as the source address for the packets.

The interface you specify with the <interface> and <number> parameters must be enabled (listed as "up" in the **show ipv6 interface** command display).

For more information about configuring tunnels, see “IPv6 Over IPv4 Tunnels” on page 8-2.

Configuring Static Neighbor Entries

In some special cases, a neighbor cannot be reached using the neighbor discovery feature. In this situation, you can add a static entry to the IPv6 neighbor discovery cache, which causes a neighbor to be reachable at all times without using neighbor discovery. (A static entry in the IPv6 neighbor discovery cache functions like a static ARP entry in IPv4.)

For example, to add a static entry for a neighbor with the IPv6 address 3001:ffe0:2678:47b and link-layer address 0004.6a2b.8641 that is reachable through Ethernet interface 3/1, enter the following command:

```
ProCurveRS(config)# ipv6 neighbor 3001:ffe0:2678:47b ethernet 3/1 0004.6a2b.8641
```

Syntax: [no] ipv6 neighbor <ipv6-address> ethernet <port> | ve <ve-number> [ethernet <port>] <link-layer-address>

The <ipv6-address> parameter specifies the address of the neighbor.

The **ethernet | ve** parameter specifies the interface through which to reach a neighbor. If you specify an Ethernet interface, specify the port number of the Ethernet interface. If you specify a VE, specify the VE number and then the Ethernet port numbers associated with the VE. The link-layer address is a 48-bit hardware address of the neighbor.

If you attempt to add an entry that already exists in the neighbor discovery cache, the software changes the already existing entry to a static entry.

To remove a static IPv6 entry from the IPv6 neighbor discovery cache, use the **no** form of this command.

Limiting the Number of Hops an IPv6 Packet Can Traverse

By default, the maximum number of hops an IPv6 packet can traverse is 64. You can change this value to between 1 – 255 hops. For example, to change the maximum number of hops to 70, you can enter the following command:

```
ProCurveRS(config)# ipv6 hop-limit 70
```

Syntax: [no] ipv6 hop-limit <number>

The number of hops can be from 1 – 255.

QoS for IPv6 Traffic

Configuring QoS for IPv6 traffic is generally the same as it is for IPv4 traffic. The QoS policies you configure on the HP device apply to both incoming IPv6 and IPv4 traffic. However, using IP ACLs to perform QoS for IPv6 traffic is not supported.

To enable QoS for IPv6 traffic, enter the following commands:

```
NI4802 Router(config)# port-priority
NI4802 Router(config)# write memory
```

```
NI4802 Router(config)# end
NI4802 Router# reload
```

Syntax: [no] port-priority

NOTE: You must save the configuration and reload the software to place the change into effect. This applies whether you are enabling QoS for IPv6 or IPv4 traffic.

The **port-priority** command globally enables QoS for IPv6 traffic on all 10/100 and 1 Gigabit interfaces. When QoS is enabled with the **port-priority** command, the device inserts a value in the internal HP header based on a combination of the following information:

- 802.1p priority
- Interface priority (if configured)
- VLAN priority (if configured)
- The first two bits in the Type of Service (ToS) header

For more information, see the Quality of Service chapter in the *Advanced Configuration and Management Guide for ProCurve 9300/9400 Series Routing Switches*.

After QoS is enabled with the **port-priority** command, you can optionally enable advanced ToS-based QoS on individual interfaces. Enabling advanced ToS-based QoS on an interface allows you to specify the trust level and packet marking used for packets received on that interface. The ***trust level*** determines the type of QoS information the device uses for performing QoS. ***Marking*** is the process of changing the packet's QoS information for the next hop.

To enable advanced ToS-based QoS on an interface, enter commands such as the following:

```
NI4802 Router(config)# int e 1
NI4802 Router(config-if-e100-1)# qos-tos
NI4802 Router(config-if-e100-1)# qos-tos trust ip-prec
NI4802 Router(config-if-e100-1)# qos-tos mark dscp
```

Syntax: [no] qos-tos

Syntax: [no] qos-tos trust cos | ip-prec | dscp

Syntax: [no] qos-tos mark cos | dscp

The commands in this example enable advanced ToS-based QoS on interface 1, set the trust level for an interface to IP Precedence, and configure the device to change the outbound packet's DSCP value to match the results of the device's QoS mapping from the specified trust level.

After you enable ToS-based QoS with the **qos-tos** command, there is no default trust level for IPv6 traffic. You must explicitly configure a trust level for IPv6 traffic. When configuring advanced ToS-based QoS, you must specify a trust level to enable DSCP marking.

Note that when advanced QoS is enabled on an interface, the configured trust level on that incoming interface determines the final priority of the packet.

For more information on configuring advanced ToS-based QoS on an interface, see the "Enhanced QoS" chapter of the *Advanced Configuration and Management Guide for ProCurve 9300/9400 Series Routing Switches*

Clearing Global IPv6 Information

You can clear the following global IPv6 information:

- Entries from the IPv6 cache.
- Entries from the IPv6 neighbor table.
- IPv6 routes from the IPv6 route table.

- IPv6 traffic statistics.
- IPv6 session flows

Clearing the IPv6 Cache

You can remove all entries from the IPv6 cache or specify an entry based on the following:

- IPv6 prefix.
- IPv6 address.
- Interface type.

For example, to remove entries for IPv6 address 2000:e0ff::1, enter the following command at the Privileged EXEC level or any of the Config levels of the CLI:

```
ProCurveRS# clear ipv6 cache 2000:e0ff::1
```

Syntax: clear ipv6 cache [<ipv6-prefix>/<prefix-length> | <ipv6-address> | ethernet <port> | tunnel <number> | ve <number>]

You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

You must specify the <ipv6-address> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

The **ethernet | tunnel | ve** parameter specifies the interfaces for which you can remove cache entries. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a VE or tunnel interface, also specify the VE or tunnel number, respectively.

Clearing IPv6 Neighbor Information

You can remove all entries from the IPv6 neighbor table or specify an entry based on the following:

- IPv6 prefix.
- IPv6 address.
- Interface type.

For example, to remove entries for Ethernet interface 3/1, enter the following command at the Privileged EXEC level or any of the CONFIG levels of the CLI:

```
ProCurveRS# clear ipv6 neighbor ethernet 3/1
```

Syntax: clear ipv6 neighbor [<ipv6-prefix>/<prefix-length> | <ipv6-address> | ethernet <port> | ve <number>]

You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

You must specify the <ipv6-address> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

The **ethernet | ve** parameter specifies the interfaces for which you can remove cache entries. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a VE, also specify the VE number.

Clearing IPv6 Routes from the IPv6 Route Table

You can clear all IPv6 routes or only those routes associated with a particular IPv6 prefix from the IPv6 route table and reset the routes.

For example, to clear IPv6 routes associated with the prefix 2000:7838::/32, enter the following command at the Privileged EXEC level or any of the Config levels of the CLI:

```
ProCurveRS# clear ipv6 route 2000:7838::/32
```

Syntax: clear ipv6 route [<ipv6-prefix>/<prefix-length>]

The <ipv6-prefix>/<prefix-length> parameter clears routes associated with a particular IPv6 prefix. You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

Clearing IPv6 Traffic Statistics

To clear all IPv6 traffic statistics (reset all fields to zero), enter the following command at the Privileged EXEC level or any of the Config levels of the CLI:

```
ProCurveRS(config)# clear ipv6 traffic
```

Syntax: clear ipv6 traffic

Deleting IPv6 Session Flows

To delete all flows from the IPv6 session cache, enter the following command:

```
ProCurveRS# clear ipv6 flows
```

Syntax: clear ipv6 flows

Displaying Global IPv6 Information

You can display output for the following global IPv6 parameters:

- IPv6 cache.
- IPv6 interfaces.
- IPv6 neighbors.
- IPv6 route table.
- Local IPv6 routers.
- IPv6 TCP connections and the status of individual connections.
- IPv6 traffic statistics.
- IPv6 session flows

Displaying IPv6 Cache Information

The IPv6 cache contains an IPv6 host table that has indices to the next hop gateway and the router interface on which the route was learned.

To display IPv6 cache information, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 cache
Total number of cache entries: 10
    IPv6 Address                        Next Hop                        Port
1   5000:2::2                           LOCAL                           tunnel 2
2   2000:4::106                         LOCAL                           ethe 3/2
3   2000:4::110                         DIRECT                          ethe 3/2
4   2002:c0a8:46a::1                    LOCAL                           ethe 3/2
5   fe80::2e0:52ff:fe99:9737            LOCAL                           ethe 3/2
6   fe80::ffff:ffff:feff:ffff           LOCAL                           loopback 2
7   fe80::c0a8:46a                      LOCAL                           tunnel 2
8   fe80::c0a8:46a                      LOCAL                           tunnel 6
9   2999::1                             LOCAL                           loopback 2
10  fe80::2e0:52ff:fe99:9700            LOCAL                           ethe 3/1
```

Syntax: show ipv6 cache [<index-number> | <ipv6-prefix>/<prefix-length> | <ipv6-address> | ethernet <port> | ve <number> | tunnel <number>]

The <index-number> parameter restricts the display to the entry for the specified index number and subsequent entries.

The <ipv6-prefix>/<prefix-length> parameter restricts the display to the entries for the specified IPv6 prefix. You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

The **ethernet | ve | tunnel** parameter restricts the display to the entries for the specified interface. The <ipv6-address> parameter restricts the display to the entries for the specified IPv6 address. You must specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a VE interface, also specify the VE number. If you specify a tunnel interface, also specify the tunnel number.

This display shows the following information:

Table 3.1: IPv6 cache information fields

This Field...	Displays...
Total number of cache entries	The number of entries in the cache table.
IPv6 Address	The host IPv6 address.
Next Hop	The next hop, which can be one of the following: • Direct – The next hop is directly connected to the router. • Local – The next hop is originated on this router. • <ipv6 address> – The IPv6 address of the next hop.
Port	The port on which the entry was learned.

Displaying IPv6 Interface Information

To display IPv6 interface information, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 interface
Routing Protocols : R - RIP  O - OSPF
Interface      Status     Routing  Global Unicast Address
Ethernet 3/3   down/down  R
Ethernet 3/5   down/down
Ethernet 3/17  up/up               2017::c017:101/64
Ethernet 3/19  up/up               2019::c019:101/64
VE 4           down/down
VE 14          up/up               2024::c060:101/64
Loopback 1     up/up               ::1/128
Loopback 2     up/up               2005::303:303/128
Loopback 3     up/up
```

Syntax: show ipv6 interface [<interface> [<port-number> I<number>]]

The <interface> parameter displays detailed information for a specified interface. For the interface, you can specify the **Ethernet**, **loopback**, **tunnel**, or **VE** keywords. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a loopback, tunnel, or VE interface, also specify the number associated with the interface.

This display shows the following information:

Table 3.2: General IPv6 interface information fields

This Field...	Displays...
Routing protocols	A one-letter code that represents a routing protocol that can be enabled on an interface.
Interface	The interface type, and the port number or number of the interface.
Status	The status of the interface. The entry in the Status field will be either “up/up” or “down/down”.
Routing	The routing protocols enabled on the interface.
Global Unicast Address	The global unicast address of the interface.

To display detailed information for a specific interface, enter a command such as the following at any CLI level:

```
ProCurveRS# show ipv6 interface ethernet 3/1
Interface Ethernet 3/1 is up, line protocol is up
  IPv6 is enabled, link-local address is fe80::2e0:52ff:fe99:97
  Global unicast address(es):
  Joined group address(es):
    ff02::9
    ff02::1:ff99:9700
    ff02::2
    ff02::1
  MTU is 1500 bytes
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 3
  ND reachable time is 30 seconds
  ND advertised reachable time is 0 seconds
  ND retransmit interval is 1 seconds
  ND advertised retransmit interval is 0 seconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  No Inbound Access List Set
  No Outbound Access List Set
  RIP enabled
```

This display shows the following information:

Table 3.3: Detailed IPv6 interface information fields

This Field...	Displays...
Interface/line protocol status	The status of interface and line protocol. If you have disabled the interface with the **disable** command, the status will be "administratively down". Otherwise, the status is either "up" or "down".
IPv6 status/link-local address	The status of IPv6. The status is either "enabled" or "disabled". Displays the link-local address, if one is configured for the interface.
Global unicast address(es)	Displays the global unicast address(es), if one or more are configured for the interface.
Joined group address(es)	The multicast address(es) that a router interface listens for and recognizes.
MTU	The setting of the maximum transmission unit (MTU) configured for the IPv6 interface. The MTU is the maximum length an IPv6 packet can have to be transmitted on the interface. If an IPv6 packet is longer than an MTU, the host that originated the packet fragments the packet and transmits its contents in multiple packets that are shorter than the configured MTU.
ICMP	The setting of the ICMP redirect parameter for the interface.
ND	The setting of the various neighbor discovery parameters for the interface.
Access List	The inbound and outbound access lists applied to the interface.
Routing protocols	The routing protocols enabled on the interface.

Displaying IPv6 Neighbor Information

You can display the IPv6 neighbor table, which contains an entry for each IPv6 neighbor with which the router exchanges IPv6 packets.

To display the IPv6 neighbor table, enter the following command at any CLI level:

```
ProCurveRS(config)# show ipv6 neighbor
Total number of Neighbor entries: 3

    IPv6 Address                            LinkLayer-Addr State Age Port    IsR
1   2000:4::110                             00e0.5291.bb37 REACH 20  ethe 3/1  1
2   fe80::2e0:52ff:fe91:bb37                00e0.5291.bb37 DELAY 1   ethe 3/2  1
3   fe80::2e0:52ff:fe91:bb40               00e0.5291.bb40 STALE 5930 ethe 3/3  1
```

Syntax: show ipv6 neighbor [<ipv6-prefix>/<prefix-length> | <ipv6-address> | <interface> [<port> |<number>]]

The <ipv6-prefix>/<prefix-length> parameters restrict the display to the entries for the specified IPv6 prefix. You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

The <ipv6-address> parameter restricts the display to the entries for the specified IPv6 address. You must specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

The <interface> parameter restricts the display to the entries for the specified router interface. For this parameter, you can specify the **Ethernet** or **VE** keywords. If you specify an Ethernet interface, also specify the port number associated with the interface. If you specify a VE interface, also specify the VE number.

This display shows the following information:

Table 3.4: IPv6 neighbor information fields

This Field...	Displays...
Total number of neighbor entries	The total number of entries in the IPv6 neighbor table.
IPv6 Address	The 128-bit IPv6 address of the neighbor.
Link-Layer Address	The 48-bit interface ID of the neighbor.
State	The current state of the neighbor. Possible states are as follows: • INCOMPLETE – Address resolution of the entry is being performed. • REACH – The forward path to the neighbor is functioning properly. • STALE – This entry has remained unused for the maximum interval. While stale, no action takes place until a packet is sent. • DELAY – This entry has remained unused for the maximum interval, and a packet was sent before another interval elapsed. • PROBE – Neighbor solicitation are transmitted until a reachability confirmation is received.
Age	The number of seconds the entry has remained unused. If this value remains unused for the number of seconds specified by the **ipv6 nd reachable-time command** (the default is 30 seconds), the entry is removed from the table.
Port	The port on which the entry was learned.
IsR	Determines if the neighbor is a Routing Switch or host: 0 – Indicates that the neighbor is a host. 1 – Indicates that the neighbor is a Routing Switch.

Displaying the IPv6 Route Table

To display the IPv6 route table, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 route
IPv6 Routing Table - 7 entries:

Type Codes:  C - Connected, S - Static, R - RIP, O - OSPF, B - BGP,

Type IPv6 Prefix                Next Hop Router           Interface  Dis/Metric
C  2000:4::/64                  ::                        ethe 3/2    0/0
S  2002::/16                    ::                        tunnel 6    1/1
S  2002:1234::/32               ::                        tunnel 6    1/1
C  2002:c0a8:46a::/64           ::                        ethe 3/2    0/0
C  2999::1/128                  ::                        loopback 2  0/0
O  2999::2/128                  fe80::2e0:52ff:fe91:bb37  ethe 3/2    110/1
C  5000:2::/64                  ::                        tunnel 2    0/0
```

Syntax: show ipv6 route [<ipv6-address> | <ipv6-prefix>/<prefix-length> | bgp | connect | ospf | rip | static | summary]

The <ipv6-address> parameter restricts the display to the entries for the specified IPv6 address. You must specify the <ipv6-address> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

The <ipv6-prefix>/<prefix-length> parameters restrict the display to the entries for the specified IPv6 prefix. You must specify the <ipv6-prefix> parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the <prefix-length> parameter as a decimal value. A slash mark (/) must follow the <ipv6-prefix> parameter and precede the <prefix-length> parameter.

The **bgp** keyword restricts the display to entries for BGP4+ routes.

The **connect** keyword restricts the display to entries for directly connected interface IPv6 routes.

The **ospf** keyword restricts the display to entries for OSPFv3 routes.

The **rip** keyword restricts the display to entries for RIPng routes.

The **static** keyword restricts the display to entries for static IPv6 routes.

The **summary** keyword displays a summary of the prefixes and different route types.

The following table lists the information displayed by the **show ipv6 route** command.

Table 3.5: IPv6 route table fields

This Field...	Displays...
Number of entries	The number of entries in the IPv6 route table.
Type	The route type, which can be one of the following: • C – The destination is directly connected to the router. • S – The route is a static route. • R – The route is learned from RIPng. • O – The route is learned from OSPFv3. • B – The route is learned from BGP4+.
IPv6 Prefix	The destination network of the route.
Next-Hop Router	The next-hop router.

Table 3.5: IPv6 route table fields (Continued)

This Field...	Displays...
Interface	The interface through which this router sends packets to reach the route's destination.
Dis/Metric	The route's administrative distance and metric value.

To display a summary of the IPv6 route table, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 route summary
IPv6 Routing Table - 7 entries:
  4 connected, 2 static, 0 RIP, 1 OSPF, 0 BGP
  Number of prefixes:
  /16: 1 /32: 1 /64: 3 /128: 2
```

The following table lists the information displayed by the **show ipv6 route summary** command:

Table 3.6: IPv6 route table summary fields

This Field...	Displays...
Number of entries	The number of entries in the IPv6 route table.
Number of route types	The number of entries for each route type.
Number of prefixes	A summary of prefixes in the IPv6 route table, sorted by prefix length.

Displaying Local IPv6 Routers

The HP device can function as an IPv6 host, instead of an IPv6 router, if you configure IPv6 addresses on its interfaces but don't enable IPv6 routing using the **ipv6 unicast-routing** command.

From the IPv6 host, you can display information about IPv6 routers to which the host is connected. The host learns about the routers through their router advertisement messages. To display information about the IPv6 routers connected to an IPv6 host, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 router
Router fe80::2e0:80ff:fe46:3431 on Ethernet 50, last update 0 min
Hops 64, Lifetime 1800 sec
Reachable time 0 msec, Retransmit time 0 msec
```

Syntax: show ipv6 router

If you configure your HP device to function as an IPv6 router (you configure IPv6 addresses on its interfaces and enable IPv6 routing using the **ipv6 unicast-routing** command) and you enter the **show ipv6 router command**, you will receive the following output:

```
No IPv6 router in table
```

Meaningful output for this command is generated for HP devices configured to function as IPv6 hosts only.

This display shows the following information:

Table 3.7: IPv6 local router information fields

This Field...	Displays...
Router <ipv6 address> on <interface> <port>	The IPv6 address for a particular router interface.
Last update	The amount of elapsed time (in minutes) between the current and previous updates received from a Routing Switch.
Hops	The default value that should be included in the Hop Count field of the IPv6 header for outgoing IPv6 packets. The hops value applies to the Routing Switch for which you are displaying information and should be followed by IPv6 hosts attached to the Routing Switch. A value of 0 indicates that the Routing Switch leaves this field unspecified.
Lifetime	The amount of time (in seconds) that the Routing Switch is useful as the default router.
Reachable time	The amount of time (in milliseconds) that a Routing Switch assumes a neighbor is reachable after receiving a reachability confirmation. The reachable time value applies to the Routing Switch for which you are displaying information and should be followed by IPv6 hosts attached to the Routing Switch. A value of 0 indicates that the Routing Switch leaves this field unspecified.
Retransmit time	The amount of time (in milliseconds) between retransmissions of neighbor solicitation messages. The retransmit time value applies to the Routing Switch for which you are displaying information and should be followed by IPv6 hosts attached to the Routing Switch. A value of 0 indicates that the Routing Switch leaves this field unspecified.

Displaying IPv6 TCP Information

You can display the following IPv6 TCP information:

- General information about each TCP connection on the Routing Switch, including the percentage of free memory for each of the internal TCP buffers.
- Detailed information about a specified TCP connection.

To display general information about each TCP connection on the Routing Switch, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 tcp connections
Local IP address:port <-> Remote IP address:port TCP state
192.168.182.110:23 <->    192.168.8.186:4933     ESTABLISHED
192.168.182.110:8218 <->  192.168.182.106:179    ESTABLISHED
192.168.182.110:8039 <->  192.168.2.119:179      SYN-SENT
192.168.182.110:8159 <->  192.168.2.102:179      SYN-SENT
2000:4::110:179 <->       2000:4::106:8222       ESTABLISHED (1440)
Total 5 TCP connections

TCP MEMORY USAGE PERCENTAGE
FREE TCB = 98 percent
FREE TCP QUEUE BUFFER = 99 percent
FREE TCP SEND BUFFER = 97 percent
FREE TCP RECEIVE BUFFER = 100 percent
FREE TCP OUT OF SEQUENCE BUFFER = 100 percent
```

Syntax: show ipv6 tcp connections

This display shows the following information:

Table 3.8: General IPv6 TCP connection fields

This Field...	Displays...
Local IP address:port	The IPv4 or IPv6 address and port number of the local router interface over which the TCP connection occurs.
Remote IP address:port	The IPv4 or IPv6 address and port number of the remote router interface over which the TCP connection occurs.

Table 3.8: General IPv6 TCP connection fields (Continued)

This Field...	Displays...
TCP state	The state of the TCP connection. Possible states include the following: • LISTEN – Waiting for a connection request. • SYN-SENT – Waiting for a matching connection request after having sent a connection request. • SYN-RECEIVED – Waiting for a confirming connection request acknowledgment after having both received and sent a connection request. • ESTABLISHED – Data can be sent and received over the connection. This is the normal operational state of the connection. • FIN-WAIT-1 – Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent. • FIN-WAIT-2 – Waiting for a connection termination request from the remote TCP. • CLOSE-WAIT – Waiting for a connection termination request from the local user. • CLOSING – Waiting for a connection termination request acknowledgment from the remote TCP. • LAST-ACK – Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request). • TIME-WAIT – Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request. • CLOSED – There is no connection state.
FREE TCB = <percentage>	The percentage of free TCP control block (TCB) space.
FREE TCB QUEUE BUFFER = <percentage>	The percentage of free TCB queue buffer space.
FREE TCB SEND BUFFER = <percentage>	The percentage of free TCB send buffer space.
FREE TCB RECEIVE BUFFER = <percentage>	The percentage of free TCB receive buffer space.
FREE TCB OUT OF SEQUENCE BUFFER = <percentage>	The percentage of free TCB out of sequence buffer space.

To display detailed information about a specified TCP connection, enter a command such as the following at any CLI level:

```
ProCurveRS# show ipv6 tcp status 2000:4::110 179 2000:4::106 8222
TCP: TCB = 0x217fc300
TCP: 2000:4::110:179 <-> 2000:4::106:8222: state: ESTABLISHED Port: 1
  Send: initial sequence number = 242365900
  Send: first unacknowledged sequence number = 242434080
  Send: current send pointer = 242434080
  Send: next sequence number to send = 242434080
  Send: remote received window = 16384
  Send: total unacknowledged sequence number = 0
  Send: total used buffers 0
  Receive: initial incoming sequence number = 740437769
  Receive: expected incoming sequence number = 740507227
  Receive: received window = 16384
  Receive: bytes in receive queue = 0
  Receive: congestion window = 1459
```

Syntax: show ipv6 tcp status <local-ip-address> <local-port-number> <remote-ip-address> <remote-port-number>

The <local-ip-address> parameter can be the IPv4 or IPv6 address of the local interface over which the TCP connection is taking place.

The <local-port-number> parameter is the local port number over which a TCP connection is taking place.

The <remote-ip-address> parameter can be the IPv4 or IPv6 address of the remote interface over which the TCP connection is taking place.

The <remote-port-number> parameter is the local port number over which a TCP connection is taking place.

This display shows the following information:

Table 3.9: Specific IPv6 TCP connection fields

This Field...	Displays...
TCB = <location>	The location of the TCB.
<local-ip-address> <local-port-number> <remote-ip-address> <remote-port-number> <state> <port>	This field provides a general summary of the following: • The local IPv4 or IPv6 address and port number. • The remote IPv4 or IPv6 address and port number. • The state of the TCP connection. For information on possible states, see Table on page 3-26. • The port numbers of the local interface.
Send: initial sequence number = <number>	The initial sequence number sent by the local router.
Send: first unacknowledged sequence number = <number>	The first unacknowledged sequence number sent by the local router.
Send: current send pointer = <number>	The current send pointer.

Table 3.9: Specific IPv6 TCP connection fields (Continued)

This Field...	Displays...
Send: next sequence number to send = <number>	The next sequence number sent by the local router.
Send: remote received window = <number>	The size of the remote received window.
Send: total unacknowledged sequence number = <number>	The total number of unacknowledged sequence numbers sent by the local router.
Send: total used buffers <number>	The total number of buffers used by the local router in setting up the TCP connection.
Receive: initial incoming sequence number = <number>	The initial incoming sequence number received by the local router.
Receive: expected incoming sequence number = <number>	The incoming sequence number expected by the local router.
Receive: received window = <number>	The size of the local router's receive window.
Receive: bytes in receive queue = <number>	The number of bytes in the local router's receive queue.
Receive: congestion window = <number>	The size of the local router's receive congestion window.

Displaying IPv6 Traffic Statistics

To display IPv6 traffic statistics, enter the following command at any CLI level:

```
ProCurveRS# show ipv6 traffic
IP6 Statistics

  36947 received, 66818 sent, 0 forwarded, 36867 delivered, 0 rawout
  0 bad vers, 23 bad scope, 0 bad options, 0 too many hdr
  0 no route, 0 can't forward, 0 redirect sent
  0 frag recv, 0 frag dropped, 0 frag timeout, 0 frag overflow
  0 reassembled, 0 fragmented, 0 ofragments, 0 can't frag
  0 too short, 0 too small, 11 not member
  0 no buffer, 66819 allocated, 21769 freed
  0 forward cache hit, 46 forward cache miss

ICMP6 Statistics
Received:
  0 dest unreach, 0 pkt too big, 0 time exceeded, 0 param prob
  2 echo req, 1 echo reply, 0 mem query, 0 mem report, 0 mem red
  0 router soli, 2393 router adv, 106 nei soli, 3700 nei adv, 0 redirect
  0 bad code, 0 too short, 0 bad checksum, 0 bad len
  0 reflect, 0 nd toomany opt, 0 badhopcount
Sent:
  0 dest unreach, 0 pkt too big, 0 time exceeded, 0 param prob
  1 echo req, 2 echo reply, 0 mem query, 0 mem report, 0 mem red
  0 router soli, 2423 router adv, 3754 nei soli, 102 nei adv, 0 redirect
  0 error, 0 can't send error, 0 too freq
Sent Errors:
  0 unreach no route, 0 admin, 0 beyond scope, 0 address, 0 no port
  0 pkt too big, 0 time exceed transit, 0 time exceed reassembly
  0 param problem header, 0 nextheader, 0 option, 0 redirect, 0 unknown

UDP Statistics
  470 received, 7851 sent, 6 no port, 0 input errors

TCP Statistics
  57913 active opens, 0 passive opens, 57882 failed attempts
  159 active resets, 0 passive resets, 0 input errors
  565189 in segments, 618152 out segments, 171337 retransmission
```

Syntax: show ipv6 traffic

This display shows the following information:

f

Table 3.10: IPv6 traffic statistics

This Field...	Displays...
IPv6 statistics	
received	The total number of IPv6 packets received by the router.
sent	The total number of IPv6 packets originated and sent by the router.
forwarded	The total number of IPv6 packets received by the router and forwarded to other routers.

Table 3.10: IPv6 traffic statistics (Continued)

This Field...	Displays...
delivered	The total number of IPv6 packets delivered to the upper layer protocol.
rawout	This information is used by HP Technical Support.
bad vers	The number of IPv6 packets dropped by the router because the version number is not 6.
bad scope	The number of IPv6 packets dropped by the router because of a bad address scope.
bad options	The number of IPv6 packets dropped by the router because of bad options.
too many hdr	The number of IPv6 packets dropped by the router because the packets had too many headers.
no route	The number of IPv6 packets dropped by the router because there was no route.
can't forward	The number of IPv6 packets the router could not forward to another router.
redirect sent	This information is used by HP Technical Support.
frag recv	The number of fragments received by the router.
frag dropped	The number of fragments dropped by the router.
frag timeout	The number of fragment timeouts that occurred.
frag overflow	The number of fragment overflows that occurred.
reassembled	The number of fragmented IPv6 packets that the router reassembled.
fragmented	The number of IPv6 packets fragmented by the router to accommodate the MTU of this router or of another device.
ofragments	The number of output fragments generated by the router.
can't frag	The number of IPv6 packets the router could not fragment.
too short	The number of IPv6 packets dropped because they are too short.
too small	The number of IPv6 packets dropped because they don't have enough data.
not member	The number of IPv6 packets dropped because the recipient is not a member of a multicast group.
no buffer	The number of IPv6 packets dropped because there is no buffer available.
forward cache miss	The number of IPv6 packets received for which there is no corresponding cache entry.
ICMP6 statistics Some ICMP statistics apply to both Received and Sent, some apply to Received only, some apply to Sent only, and some apply to Sent Errors only.	
Applies to Received and Sent	
dest unreach	The number of Destination Unreachable messages sent or received by the router.
pkt too big	The number of Packet Too Big messages sent or received by the router.

Table 3.10: IPv6 traffic statistics (Continued)

This Field...	Displays...
time exceeded	The number of Time Exceeded messages sent or received by the router.
param prob	The number of Parameter Problem messages sent or received by the router.
echo req	The number of Echo Request messages sent or received by the router.
echo reply	The number of Echo Reply messages sent or received by the router.
mem query	The number of Group Membership Query messages sent or received by the router.
mem report	The number of Membership Report messages sent or received by the router.
mem red	The number of Membership Reduction messages sent or received by the router.
router soli	The number of Router Solicitation messages sent or received by the router.
router adv	The number of Router Advertisement messages sent or received by the router.
nei soli	The number of Neighbor Solicitation messages sent or received by the router.
nei adv	The number of Router Advertisement messages sent or received by the router.
redirect	The number of redirect messages sent or received by the router.
Applies to Received Only	
bad code	The number of Bad Code messages received by the router.
too short	The number of Too Short messages received by the router.
bad checksum	The number of Bad Checksum messages received by the router.
bad len	The number of Bad Length messages received by the router.
nd toomany opt	The number of Neighbor Discovery Too Many Options messages received by the router.
badhopcount	The number of Bad Hop Count messages received by the router.
Applies to Sent Only	
error	The number of Error messages sent by the router.
can’t send error	The number of times the node encountered errors in ICMP error messages.
too freq	The number of times the node has exceeded the frequency of sending error messages.
Applies to Sent Errors Only	
unreach no route	The number of Unreachable No Route errors sent by the router.
admin	The number of Admin errors sent by the router.
beyond scope	The number of Beyond Scope errors sent by the router.
address	The number of Address errors sent by the router.
no port	The number of No Port errors sent by the router.

Table 3.10: IPv6 traffic statistics (Continued)

This Field...	Displays...
pkt too big	The number of Packet Too Big errors sent by the router.
time exceed transit	The number of Time Exceed Transit errors sent by the router.
time exceed reassembly	The number of Time Exceed Reassembly errors sent by the router.
param problem header	The number of Parameter Problem Header errors sent by the router.
nextheader	The number of Next Header errors sent by the router.
option	The number of Option errors sent by the router.
redirect	The number of Redirect errors sent by the router.
unknown	The number of Unknown errors sent by the router.
UDP statistics	
received	The number of UDP packets received by the router.
sent	The number of UDP packets sent by the router.
no port	The number of UDP packets dropped because the packet did not contain a valid UDP port number.
input errors	This information is used by HP Technical Support.
TCP statistics	
active opens	The number of TCP connections opened by the router by sending a TCP SYN to another device.
passive opens	The number of TCP connections opened by the router in response to connection requests (TCP SYNs) received from other devices.
failed attempts	This information is used by HP Technical Support.
active resets	The number of TCP connections the router reset by sending a TCP RESET message to the device at the other end of the connection.
passive resets	The number of TCP connections the router reset because the device at the other end of the connection sent a TCP RESET message.
input errors	This information is used by HP Technical Support.
in segments	The number of TCP segments received by the router.
out segments	The number of TCP segments sent by the router.
retransmission	The number of segments that the router retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment.

Displaying IPv6 Session Flows

If you want to display the contents of an IPv6 session cache, enter the following command:

```
ProCurveRS# show ipv6 flows
```

Syntax: show ipv6 flows [<source-ipv6-prefix/prefix-length> | any | host <source-ipv6_address> <destination-ipv6-prefix/prefix-length> | any | host <destination-ipv6-address>]

If you do not specify a source or destination, all IPv6 flows are displayed.

Enter a value for <source-ipv6-prefix>/<prefix-length> or <destination-ipv6-prefix>/<prefix-length> to specify a source or destination prefix and prefix length that a flow must match to be included in the display.

Enter **any** for source or destination if a flow can have any source or any destination to be included in the display.

The **host** <source-ipv6-address> and **host** <destination-ipv6-address> parameters allow you specify a source or destination host IPv6 address that a flow must match to be included in the display.

EXAMPLES:

To show all IPv6 flows, enter the following command:

```
ProCurveRS# show ipv6 flows
```

To show all IPv6 flows with any IPv6 source and any IPv6 destination addresses, enter the following command:

```
ProCurveRS# show ipv6 flows any any
```

To show all IPv6 flows that match the source prefix 4000::/16 and any destination address, enter the following command:

```
ProCurveRS# show ipv6 flows 4000::/16 any
```

To show all IPv6 flows that have any source address but only a destination address of host 5020::30, enter the following command:

```
ProCurveRS# show ipv6 flows any host 5020::30
```

To show all IPv6 flows that have the source address of host 4050::30 and the destination address of host 5020::30, enter the following command:

```
ProCurveRS# show ipv6 flows host 4050::30 host 5020::30
```

The following is an example of what is displayed when you enter the **show ipv6 flows** command:

```
ProCurveRS# show ipv6 flows

ipv6 flows count: 6
A:Ack D:Deny E:Estab F:Fin P:Psh Pe:Permit R:Rst U:urg Fr:Fragment
Sr:SRouted
SourceAddress                          DestinationAddress
  Protocol SrcPort/IcmpType DestPort/IcmpCode Dscp FlowLabel  Flags     Age
3001::3                                3020::160
 icmp      128             0                 0    0          Pe        4
3001::3                                3020::160
 tcp       telnet          3456              0    0          DAR       3
3001::3                                3020::160
 tcp       telnet          3456              0    0          DAS       3
3001::3                                3020::160
 icmp      129             0                 0    0          Pe        8
3001::3                                3020::160
 tcp       3456            telnet            0    0          DAR       9
3001::3                                3020::165
 icmp      128             0                 0    0          Pe        4
```

The first line (ipv6 flows count) shows the number of flows included on the display.

The next line defines the flags used in the display.

Information for each flow on the display appears on two lines in the following sequence:

- Source Address – Source address of the flow.
- Destination Address – Destination address of the flow.

- Protocol – Protocol in the flow.
- SrcPort/IcmpType – Either the source TCP/UDP port or the ICMP type of the flow.
- DestPort/IcmpCode – Either the destination TCP/UDP port or the ICMP code of the flow.
- Dscp – DSCP value in the flow.
- FlowLabel – Value in the flow label field of the IPv6 packet header.
- Flags – Status of the flow, which can be a combination of different flag types. For example, DAR means the flow was denied (D), acknowledged (A), and reset (R).
- Age – Age of the flow.

NOTE: The life of an idle flow is 50 seconds.