C H A P T E R 36

Configuring IPv6 Host Functions

This chapter describes how to configure IPv6host functions on the switch.

For information about configuring IPv4 unicast routing, see Chapter 35, "Configuring IP Unicast Routing." For information about configuring IPv6 Multicast Listener Discovery (MLD) snooping, see Chapter 37, "Configuring IPv6 MLD Snooping."

To enable dual stack environments (supporting both IPv4 and IPv6), you must you must configure the switch to use the a dual IPv4 and IPv6 switch database management (SDM) template. See the "Dual IPv4 and IPv6 Protocol Stacks" section on page 36-5. For more information about SDM templates, see Chapter 6, "Configuring SDM Templates."

> **Note** For complete syntax and usage information for the commands used in this chapter, see the Cisco IOS documentation referenced in the procedures

This chapter consists of these sections:

- "Understanding IPv6" section on page 36-1
- "Configuring IPv6" section on page 36-7
- "Displaying IPv6" section on page 36-12

Understanding IPv6

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

For information about how Cisco Systems implements IPv6, go to this URL:

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

For information about IPv6 and other features in this chapter

- See the *Cisco IOS IPv6 Configuration Library* at this URL:

  http://www.cisco.com/en/US/docs/ios/12_2t/ipv6/ipv6_c.html

- Use the Search field to locate the Cisco IOS software documentation. For example, if you want information about static routes, you can enter *Implementing Static Routes for IPv6* in the search field to get this document about static routes:

  http://www.cisco.com/en/US/docs/ios/12_2t/ipv6/SA_Stat6.html

This section describes IPv6 implementation on the switch. These sections are included:

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, anycast addresses, or multicast addresses.

The IPv6 128-bit addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons in the format: n:n:n:n:n:n:n:n. This is an example of an IPv6 address:

2031:0000:130F:0000:0000:09C0:080F:130B

For easier implementation, leading zeros in each field are optional. This is the same address without leading zeros:

2031:0:130F:0:0:9C0:80F:130B

You can also use two colons (::) to represent successive hexadecimal fields of zeros, but you can use this short version only once in each address:

2031:0:130F::09C0:080F:130B

For more information about IPv6 address formats, address types, and the IPv6 packet header, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

In the "Implementing Addressing and Basic Connectivity" chapter, these sections apply to the switch:

- IPv6 Address Formats
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Unicast Host Features

These sections describe the IPv6 protocol features supported by the switch:

Support on the switch includes expanded address capability, header format simplification, improved support of extensions and options, and hardware parsing of the extension header. The switch supports hop-by-hop extension header packets, which are routed or bridged in software.

128-Bit Wide Unicast Addresses

The switch supports aggregatable global unicast addresses and link-local unicast addresses. It does not support site-local unicast addresses.

- Aggregatable global unicast addresses are IPv6 addresses from the aggregatable global unicast prefix. The address structure enables strict aggregation of routing prefixes and limits the number of routing table entries in the global routing table. These addresses are used on links that are aggregated through organizations and eventually to the Internet service provider.

  These addresses are defined by a global routing prefix, a subnet ID, and an interface ID. Current global unicast address allocation uses the range of addresses that start with binary value 001 (2000::/3). Addresses with a prefix of 2000::/3(001) through E000::/3(111) must have 64-bit interface identifiers in the extended unique identifier (EUI)-64 format.

- Link local unicast addresses can be automatically configured on any interface by using the link-local prefix FE80::/10(1111 1110 10) and the interface identifier in the modified EUI format. Link-local addresses are used in the neighbor discovery protocol and the stateless autoconfiguration process. Nodes on a local link use link-local addresses and do not require globally unique addresses to communicate. IPv6 routers do not forward packets with link-local source or destination addresses to other links.

For more information, see the section about IPv6 unicast addresses in the "Implementing IPv6 Addressing and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DNS for IPv6

IPv6 supports Domain Name System (DNS) record types in the DNS name-to-address and address-to-name lookup processes. The DNS AAAA resource record types support IPv6 addresses and are equivalent to an A address record in IPv4. The switch supports DNS resolution for IPv4 and IPv6.

ICMPv6

The Internet Control Message Protocol (ICMP) in IPv6 generates error messages, such as ICMP destination unreachable messages, to report errors during processing and other diagnostic functions. In IPv6, ICMP packets are also used in the neighbor discovery protocol and path MTU discovery.

Neighbor Discovery

The switch supports NDP for IPv6, a protocol running on top of ICMPv6, and static neighbor entries for IPv6 stations that do not support NDP. The IPv6 neighbor discovery process uses ICMP messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), to verify the reachability of the neighbor, and to keep track of neighboring routers.

The switch supports ICMPv6 redirect for routes with mask lengths less than 64 bits. ICMP redirect is not supported for host routes or for summarized routes with mask lengths greater than 64 bits.

Neighbor discovery throttling ensures that the switch CPU is not unnecessarily burdened while it is in the process of obtaining the next hop forwarding information to route an IPv6 packet. The switch drops any additional IPv6 packets whose next hop is the same neighbor that the switch is actively trying to resolve. This drop avoids further load on the CPU.

Default Router Preference

The switch supports IPv6 default router preference (DRP), an extension in router advertisement messages. DRP improves the ability of a host to select an appropriate router, especially when the host is multihomed and the routers are on different links. The switch does not support the Route Information Option in RFC 4191.

An IPv6 host maintains a default router list from which it selects a router for traffic to offlink destinations. The selected router for a destination is then cached in the destination cache. NDP for IPv6 specifies that routers that are reachable or probably reachable are preferred over routers whose reachability is unknown or suspect. For reachable or probably reachable routers, NDP can either select the same router every time or cycle through the router list. By using DRP, you can configure an IPv6 host to prefer one router over another, provided both are reachable or probably reachable.

For more information about DRP for IPv6, see the "Implementing IPv6 Addresses and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

The switch uses stateless autoconfiguration to manage link, subnet, and site addressing changes, such as management of host and mobile IP addresses. A host autonomously configures its own link-local address, and booting nodes send router solicitations to request router advertisements for configuring interfaces.

For more information about autoconfiguration and duplicate address detection, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Applications

The switch has IPv6 support for these applications:

- Ping, traceroute, Telnet, TFTP, and FTP
- Secure Shell (SSH) over an IPv6 transport
- HTTP server access over IPv6 transport
- DNS resolver for AAAA over IPv4 transport
- Cisco Discovery Protocol (CDP) support for IPv6 addresses

For more information about managing these applications, see the "Managing Cisco IOS Applications over IPv6" chapter and the "Implementing IPv6 Addressing and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Dual IPv4 and IPv6 Protocol Stacks

You must use the dual IPv4 and IPv6 template to allocate ternary content addressable memory (TCAM) usage to both IPv4 and IPv6 protocols.

Figure 36-1 shows a router forwarding both IPv4 and IPv6 traffic through the same interface, based on the IP packet and destination addresses.

Figure 36-1   Dual IPv4 and IPv6 Support on an Interface
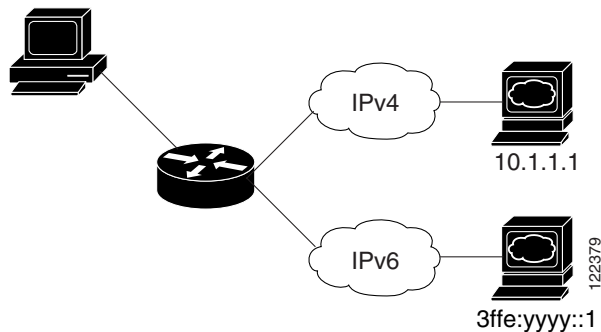
Use the dual IPv4 and IPv6 switch database management (SDM) template to enable dual stack environments (supporting both IPv4 and IPv6). For more information about the dual IPv4 and IPv6 SDM template, see Chapter 6, "Configuring SDM Templates."

The dual Pv4 and IPv6 templates allow the switch to be used in dual stack environments.

- If you try to configure IPv6 without first selecting a dual IPv4 and IPv6 template, a warning message appears.
- In IPv4-only environments, the switch applies IPv4 QoS and ACLs in hardware. IPv6 packets are not supported.
- In dual IPv4 and IPv6 environments, the switch applies IPv4 QoS and ACLs in hardware.
- IPv6 QoS and ACLs are not supported.
- If you do not plan to use IPv6, do not use the dual stack template because this template results in less TCAM capacity for each resource.

For more information about IPv4 and IPv6 protocol stacks, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Static Routes for IPv6

Static routes are manually configured and define an explicit route between two networking devices. Static routes are useful for smaller networks with only one path to an outside network or to provide security for certain types of traffic in a larger network.

For more information about static routes, see the "Implementing Static Routes for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

SNMP and Syslog Over IPv6

To support both IPv4 and IPv6, IPv6 network management requires both IPv6 and IPv4 transports. Syslog over IPv6 supports address data types for these transports.

SNMP and syslog over IPv6 provide these features:

- Support for both IPv4 and IPv6
- IPv6 transport for SNMP and to modify the SNMP agent to support traps for an IPv6 host
- SNMP- and syslog-related MIBs to support IPv6 addressing
- Configuration of IPv6 hosts as trap receivers

For support over IPv6, SNMP modifies the existing IP transport mapping to simultaneously support IPv4 and IPv6. These SNMP actions support IPv6 transport management:

- Opens User Datagram Protocol (UDP) SNMP socket with default settings
- Provides a new transport mechanism called *SR_IPV6_TRANSPORT*
- Sends SNMP notifications over IPv6 transport
- Supports SNMP-named access lists for IPv6 transport
- Supports SNMP proxy forwarding using IPv6 transport
- Verifies SNMP Manager feature works with IPv6 transport

For information on SNMP over IPv6, including configuration procedures, see the "Managing Cisco IOS Applications over IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

For information about syslog over IPv6, including configuration procedures, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

HTTP(S) Over IPv6

The HTTP client sends requests to both IPv4 and IPv6 HTTP servers, which respond to requests from both IPv4 and IPv6 HTTP clients. URLs with literal IPv6 addresses must be specified in hexadecimal using 16-bit values between colons.

The accept socket call chooses an IPv4 or IPv6 address family. The accept socket is either an IPv4 or IPv6 socket. The listening socket continues to listen for both IPv4 and IPv6 signals that indicate a connection. The IPv6 listening socket is bound to an IPv6 wildcard address.

The underlying TCP/IP stack supports a dual-stack environment. HTTP relies on the TCP/IP stack and the sockets for processing network-layer interactions.

Basic network connectivity (**ping**) must exist between the client and the server hosts before HTTP connections can be made.

For more information, see the "Managing Cisco IOS Applications over IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring IPv6

These sections contain this IPv6 forwarding configuration information:

- Default IPv6 Configuration, page 36-7
- Configuring IPv6 Addressing and Enabling IPv6 Host, page 36-7
- Configuring Default Router Preference, page 36-9
- Configuring IPv6 ICMP Rate Limiting, page 36-9
- Configuring Static Routes for IPv6, page 36-10

> **Note** Before you configure IPv6 on a switch, you must enter the **sdm prefer** {**dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**} global configuration command and reload the switch to configure a dual template.

Default IPv6 Configuration

Table 36-1 shows the default IPv6 configuration.

Table 36-1   Default IPv6 Configuration

Feature	Default Setting
SDM template	Default
IPv6 addresses	None configured

Configuring IPv6 Addressing and Enabling IPv6 Host

This section describes how to assign IPv6 addresses to individual Layer 3 interfaces and to globally forward IPv6 traffic on the switch.

Before configuring IPv6 on the switch, consider these guidelines:

- Be sure to select a dual IPv4 and IPv6 SDM template.
- In the **ipv6 address** interface configuration command, you must enter the *ipv6-address* and *ipv6-prefix* variables with the address specified in hexadecimal using 16-bit values between colons. The *prefix-length* variable (preceded by a slash [/]) is a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

To forward IPv6 traffic on an interface, you must configure a global IPv6 address on that interface. Configuring an IPv6 address on an interface automatically configures a link-local address and activates IPv6 for the interface. The configured interface automatically joins these 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 (this address is used in the neighbor discovery process.)
- all-nodes link-local multicast group FF02::1
- all-routers link-local multicast group FF02::2

For more information about configuring IPv6, see the "Implementing Addressing and Basic Connectivity for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Beginning in privileged EXEC mode, follow these steps to assign an IPv6 address to a Layer 3 interface and enable IPv6 forwarding:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**sdm prefer dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**}	Select an SDM template that supports IPv4 and IPv6. • **default**—Set the switch to the default template to balance system resources. • **routing**—Set the switch to the routing template to support IPv4 and IPv6 traffic. • **vlan**—Maximize VLAN configuration on the switch.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**reload**	Reload the operating system.
Step 5	**configure terminal**	Enter global configuration mode after the switch reloads.
Step 6	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure. The interface can be a physical interface, a switch virtual interface (SVI), or a Layer 3 EtherChannel.
Step 7	**no switchport**	Remove the interface from Layer 2 configuration mode (if it is a physical interface).
Step 8	**ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **ipv6 address** *ipv6-address* **link-local** or **ipv6 enable**	Specify a global IPv6 address with an extended unique identifier (EUI) in the low-order 64 bits of the IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. This enables IPv6 processing on the interface. Specify a link-local address on the interface to be used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface. This command enables IPv6 processing on the interface. Automatically configure an IPv6 link-local address on the interface, and enable the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.
Step 9	**exit**	Return to global configuration mode.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show ipv6 interface** *interface-id*	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IPv6 address from an interface, use the **no ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **no ipv6 address** *ipv6-address* **link-local** interface configuration command. To remove all manually configured IPv6 addresses from an interface, use the **no ipv6 address** interface configuration command without arguments. To disable IPv6 processing on an interface that has not been explicitly configured with an IPv6 address, use the **no ipv6 enable** interface configuration command. This example shows how to enable IPv6 host on an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/11
```

```
Switch(config-if)# no switchport
Switch(config-if)# ipv6 enable
Switch(config-if)# exit
```

Configuring Default Router Preference

Router advertisement messages are sent with the default router preference (DRP) configured by the **ipv6 nd router-preference** interface configuration command. If no DRP is configured, RAs are sent with a medium preference.

A DRP is useful when two routers on a link might provide equivalent, but not equal-cost routing, and policy might dictate that hosts should prefer one of the routers.

Beginning in privileged EXEC mode, follow these steps to configure a DRP for a router on an interface.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the Layer 3 interface on which you want to specify the DRP.
Step 3	**ipv6 nd router-preference** {**high** | **medium** | **low**}	Specify a DRP for the router on the switch interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ipv6 interface**	Verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ipv6 nd router-preference** interface configuration command to disable an IPv6 DRP.

This example shows how to configure a DRP of *high* for the router on an interface.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ipv6 nd router-preference high
Switch(config-if)# end
```

For more information about configuring DRP for IPv6, see the "Implementing IPv6 Addresses and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring IPv6 ICMP Rate Limiting

ICMP rate limiting is enabled by default with a default interval between error messages of 100 milliseconds and a bucket size (maximum number of tokens to be stored in a bucket) of 10.

Beginning in privileged EXEC mode, follow these steps to change the ICMP rate-limiting parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 icmp error-interval** *interval* [*bucketsize*]	Configure the interval and bucket size for IPv6 ICMP error messages: • *interval*—The interval (in milliseconds) between tokens being added to the bucket. The range is from 0 to 2147483647 milliseconds. • *bucketsize*—(Optional) The maximum number of tokens stored in the bucket. The range is from 1 to 200.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ipv6 interface** [*interface-id*]	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default configuration, use the **no ipv6 icmp error-interval** global configuration command.

This example shows how to configure an IPv6 ICMP error message interval of 50 milliseconds and a bucket size of 20 tokens.

```
Switch(config)#ipv6 icmp error-interval 50 20
```

Configuring Static Routes for IPv6

Before configuring a static IPv6 route, you must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on at least one Layer 3 interface by configuring an IPv6 address on the interface.

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 static route:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address*	*interface-id* [*ipv6-address*]} [*administrative distance*]	Configure a static IPv6 route. • *ipv6-prefix*—The IPv6 network that is the destination of the static route. It can also be a hostname when static host routes are configured. • */prefix length*—The length of the IPv6 prefix. A decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. • *ipv6-address*—The IPv6 address of the next hop that can be used to reach the specified network. The IPv6 address of the next hop need not be directly connected; recursion is done to find the IPv6 address of the directly connected next hop. The address must be in the form documented in RFC 2373, specified in hexadecimal using 16-bit values between colons. • *interface-id*—Specify direct static routes from point-to-point and broadcast interfaces. With point-to-point interfaces, there is no need to specify the IPv6 address of the next hop. With broadcast interfaces, you should always specify the IPv6 address of the next hop, or ensure that the specified prefix is assigned to the link, specifying a link-local address as the next hop. You can optionally specify the IPv6 address of the next hop to which packets are sent. **Note** You must specify an *interface-id* when using a link-local address as the next hop (the link-local next hop must also be an adjacent router). • *administrative distance*—(Optional) An administrative distance. The range is 1 to 254; the default value is 1, which gives static routes precedence over any other type of route except connected routes. To configure a floating static route, use an administrative distance greater than that of the dynamic routing protocol.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	show ipv6 static [*ipv6-address* | *ipv6-prefix/prefix length*] [**interface** *interface-id*] [**recursive**] [**detail**] or show ipv6 route static [*updated*]	Verify your entries by displaying the contents of the IPv6 routing table. • **interface** *interface-id*—(Optional) Display only those static routes with the specified interface as an egress interface. • **recursive**—(Optional) Display only recursive static routes. The **recursive** keyword is mutually exclusive with the **interface** keyword, but it can be used with or without the IPv6 prefix included in the command syntax. • **detail**—(Optional) Display this additional information: – For valid recursive routes, the output path set, and maximum resolution depth. – For invalid routes, the reason why the route is not valid.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

To remove a configured static route, use the **no ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* \| *interface-id* [*ipv6-address*]} [*administrative distance*] global configuration command.

This example shows how to configure a floating static route to an interface with an administrative distance of 130:

```
Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet0/1 130
```

For more information about configuring static IPv6 routing, see the "Implementing Static Routes for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Displaying IPv6

For complete syntax and usage information on these commands, see the Cisco IOS command reference publications.

Table 36-2 shows the privileged EXEC commands for monitoring IPv6 on the switch.

Table 36-2   Commands for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Display a summary of access lists.
show ipv6 interface *interface-id*	Display IPv6 interface status and configuration.
show ipv6 mtu	Display IPv6 MTU per destination cache.
show ipv6 neighbors	Display IPv6 neighbor cache entries.
show ipv6 prefix-list	Display a list of IPv6 prefix lists.
show ipv6 static	Display IPv6 static routes.
show ipv6 traffic	Display IPv6 traffic statistics.

Table 36-3 shows the privileged EXEC commands for displaying information about IPv4 and IPv6 address types.

Table 36-3   Commands for Displaying IPv4 and IPv6 Address Types

Command	Purpose
show ip http server history	Display the previous 20 connections to the HTTP server, including the IP address accessed and the time when the connection was closed.
show ip http server connection	Display the current connections to the HTTP server, including the local and remote IP addresses being accessed.
show ip http client connection	Display the configuration values for HTTP client connections to HTTP servers.
show ip http client history	Display a list of the last 20 requests made by the HTTP client to the server.

This is an example of the output from the **show ipv6 interface** privileged EXEC command:

```
Switch# show ipv6 interface
Vlan1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```

This is an example of the output from the **show ipv6 protocols** privileged EXEC command:

```
Switch# show ipv6 protocols
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "static"
IPv6 Routing Protocol is "rip fer"
  Interfaces:
    Vlan6
    GigabitEthernet0/4
    GigabitEthernet0/11
    GigabitEthernet0/12
  Redistribution:
    None
```

This is an example of the output from the **show ipv6 static** privileged EXEC command:

```
Switch# show ipv6 static
IPv6 Static routes
Code: * - installed in RIB
* ::/0 via nexthop 3FFE:C000:0:7::777, distance 1
```

This is an example of the output from the **show ipv6 neighbor** privileged EXEC command:

```
Switch# show ipv6 neighbors
IPv6 Address                          Age Link-layer Addr State Interface
3FFE:C000:0:7::777                      - 0007.0007.0007  REACH Vl7
3FFE:C101:113:1::33                     - 0000.0000.0033  REACH Fa1/0/13
```

This is an example of the output from the **show ipv6 traffic** privileged EXEC command.

```
Switch# show ipv6 traffic
IPv6 statistics:
  Rcvd:  1 total, 1 local destination
         0 source-routed, 0 truncated
         0 format errors, 0 hop count exceeded
         0 bad header, 0 unknown option, 0 bad source
         0 unknown protocol, 0 not a router
         0 fragments, 0 total reassembled
         0 reassembly timeouts, 0 reassembly failures
  Sent:  36861 generated, 0 forwarded
         0 fragmented into 0 fragments, 0 failed
         0 encapsulation failed, 0 no route, 0 too big
         0 RPF drops, 0 RPF suppressed drops
  Mcast: 1 received, 36861 sent

ICMP statistics:
  Rcvd: 1 input, 0 checksum errors, 0 too short
        0 unknown info type, 0 unknown error type
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        1 router solicit, 0 router advert, 0 redirects
        0 neighbor solicit, 0 neighbor advert
  Sent: 10112 output, 0 rate-limited
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        0 router solicit, 9944 router advert, 0 redirects
        84 neighbor solicit, 84 neighbor advert

UDP statistics:
  Rcvd: 0 input, 0 checksum errors, 0 length errors
        0 no port, 0 dropped
  Sent: 26749 output

TCP statistics:
  Rcvd: 0 input, 0 checksum errors
  Sent: 0 output, 0 retransmitted
```