Configuring IPv6 Unicast Routing

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Configuring IPv6 Unicast Routing

This chapter describes how to configure IPv6 unicast routing on the switch.

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:

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*.
- Use the Search field on Cisco.com 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 learn about static routes.

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, or anycast 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 http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xe-3e/ip6b-xe-3e-book.html of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

In the "Implementing Addressing and Basic Connectivity" chapter, these sections apply to the Catalyst 2960, 2960-S, 2960-C, 2960-X, 2960-CX and 3560-CX switches:

- IPv6 Address Formats
- IPv6 Address Type: Multicast
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Unicast Routing Features

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

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 (NDP) 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.

Path MTU Discovery for IPv6 Unicast

The switch supports advertising the system maximum transmission unit (MTU) to IPv6 nodes and path MTU discovery. Path MTU discovery allows a host to dynamically discover and adjust to differences in the MTU size of every link along a given data path. In IPv6, if a link along the path is not large enough to accommodate the packet size, the source of the packet handles the fragmentation.

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 configuring DRP for IPv6, see the *Configuring Default Router Preference* section.

For more information about DRP for IPv6, see 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, and Telnet
- 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 *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DHCP for IPv6 Address Assignment

DHCPv6 enables DHCP servers to pass configuration parameters, such as IPv6 network addresses, to IPv6 clients. The address assignment feature manages non-duplicate address assignment in the correct prefix based on the network where the host is connected. Assigned addresses can be from one or multiple prefix pools. Additional options, such as default domain and DNS name-server address, can be passed back to the client. Address pools can be assigned for use on a specific interface, on multiple interfaces, or the server can automatically find the appropriate pool.

For configuring DHCP for IPv6, see the *Configuring DHCP for IPv6 Address Assignment* section.

For more information about configuring the DHCPv6 client, server, or relay agent functions, see the *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.

Configuring Static Routing for IPv6 (CLI)

For configuring static routes for IPv6, see the *Configuring Static Routing for IPv6* section.

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

RIP for IPv6

Routing Information Protocol (RIP) for IPv6 is a distance-vector protocol that uses hop count as a routing metric. It includes support for IPv6 addresses and prefixes and the all-RIP-routers multicast group address FF02::9 as the destination address for RIP update messages.

For configuring RIP for IPv6, see the *Configuring RIP for IPv6* section.

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

OSPF for IPv6

The switch running the feature set supports Open Shortest Path First (OSPF) for IPv6, a link-state protocol for IP. For more information, see*Cisco IOS IPv6 Configuration Library* on Cisco.com.

OSPFv3 Graceful Restart

OSPFv3 feature allows nonstop data forwarding along known routes while the OSPFv3 routing protocol information is restored. A switch uses graceful restart either in restart mode (for a graceful-restart-capable switch) or in helper mode (for a graceful-restart-aware switch).

To use the graceful restart function, a switch must be in high-availability stateful switchover (SSO) mode (dual route processor). A switch capable of graceful restart uses it when these failures occur:

- A route processor failure that results in changeover to the standby route processor
- A planned route processor changeover to the standby route processor

The graceful restart feature requires that neighboring switches be graceful-restart aware.

For more information, see the Implementing OSPF for IPv6 chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Fast Convergence: LSA and SPF Throttling

The OSPFv3 link-state advertisements (LSA) and shortest path first (SPF) throttling feature provides a dynamic method to slow down link-state advertisement updates in OSPFv3 during times of network instability. This feature also allows faster OSPFv3 convergence by providing LSA rate limiting in milliseconds.

OSPFv3 previously used static timers for rate-limiting SPF calculation and LSA generation. Although these timers are configurable, the values are specified in seconds, which poses a limitation on OSPFv3 convergence. LSA and SPF throttling achieves subsecond convergence by providing a more sophisticated SPF and LSA rate-limiting method can react quickly to changes and also provide stability and protection during prolonged periods of instability.

Authentication Support with IPsec

To ensure that OSPF for IPv6 (OSPFv3) packets are not altered and resent to the switch, OSPFv3 packets must be authenticated. OSPFv3 uses the IPsec secure socket API to add authentication to OSPFv3 packets. This API has been extended to provide support for IPv6.

OSPFv3 requires the use of IPsec to enable authentication. Crypto images are required to use authentication, because only crypto images include the IPsec API needed for use with OSPFv3.

Configuring HSRP for IPv6

HSRP provides routing redundancy for routing IPv6 traffic not dependent on the availability of any single router. IPv6 hosts learn of available routers through IPv6 neighbor discovery router advertisement messages. These messages are multicast periodically or are solicited by hosts.

An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number and a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address. Periodic messages

are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active. These messages stop after a final one is sent when the group leaves the active state.

Note When configuring HSRP for IPv6, you must enable HSRP version 2 (HSRPv2) on the interface.

EIGRP IPv6

Switches running the IP services feature set support the Enhanced Interior Gateway Routing Protocol (EIGRP) for IPv6. It is configured on the interfaces on which it runs and does not require a global IPv6 address.

Note Switches running the IP base feature set do not support any IPv6 EIGRP features, including IPv6 EIGRP stub routing.

Before running, an instance of EIGRP IPv6 requires an implicit or explicit router ID. An implicit router ID is derived from a local IPv4 address, so any IPv4 node always has an available router ID. However, EIGRP IPv6 might be running in a network with only IPv6 nodes and therefore might not have an available IPv4 router ID.

For more information about EIGRP for IPv6, see the "Implementing EIGRP 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.

Simple Network Management Protocol (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.

Unsupported IPv6 Unicast Routing Features

The switch does not support these IPv6 features:

- IPv6 packets destined to site-local addresses
- Tunneling protocols, such as IPv4-to-IPv6 or IPv6-to-IPv4
- The switch as a tunnel endpoint supporting IPv4-to-IPv6 or IPv6-to-IPv4 tunneling protocols

IPv6 Feature Limitations

Because IPv6 is implemented in switch hardware, some limitations occur due to the IPv6 compressed addresses in the hardware memory. These hardware limitations result in some loss of functionality and limits some features.

These are feature limitations.

- The switch cannot forward SNAP-encapsulated IPv6 packets in hardware. They are forwarded in software.
- The switch cannot apply QoS classification on source-routed IPv6 packets in hardware.

Configuring IPv6

Default IPv6 Configuration

Table 1: Default IPv6 Configuration

Feature	Default Setting
SDM template	Advance desktop. Default is advanced template Default

Feature	Default Setting
IPv6 addresses	None configured

Configuring IPv6 Addressing and Enabling IPv6 Routing

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 routing, 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 routing:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode after the switch reloads.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure. The interface can be a physical interface, a switch virtual interface (SVI), or a Layer 3 EtherChannel.
Step 3	**no switchport** **Example:**	Removes the interface from Layer 2 configuration mode (if it is a physical interface).

	Command or Action	Purpose
	`Switch(config-if)# no switchport`	

Step 4	Use one of the following:
	• **ipv6 address** *ipv6-prefix/prefix length* **eui-64**
	• **ipv6 address** *ipv6-address/prefix length*
	• **ipv6 address** *ipv6-address* **link-local**
	• **ipv6 enable**
	Example:
	`Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64`
	`Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64`
	`Switch(config-if)# ipv6 address 2001:0DB8:c18:1:: link-local`
	`Switch(config-if)# ipv6 enable`
		• Specifies 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.
		• Manually configures an IPv6 address on the interface.
		• Specifies 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 configures an IPv6 link-local address on the interface, and enables the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.

Step 5	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 6	**ip routing** **Example:** `Switch(config)# ip routing`	Enables IP routing on the switch.
Step 7	**ipv6 unicast-routing** **Example:** `Switch(config)# ipv6 unicast-routing`	Enables forwarding of IPv6 unicast data packets.
Step 8	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ipv6 interface** *interface-id* **Example:**	Verifies your entries.

	Command or Action	Purpose
	`Switch# show ipv6 interface gigabitethernet 1/0/1`	
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring IPv6 Addressing and Enabling IPv6 Routing: Example, on page 51

Configuring First Hop Security in IPv6

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):
  - A physical port with an FHS policy attached cannot join an EtherChannel group.
  - An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.
- By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:
  - Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.
  - Configure a snooping policy with a lower security-level, for example glean or inspect. However; configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.

- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

  This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.

- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.

- IPv6 DHCP Guard—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.

- IPv6 Source Guard—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

  A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

  To debug source-guard packets, use the debug ipv6 snooping source-guard privileged EXEC command.

  The following restrictions apply:

  - An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.
  - When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.
  - An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.
  - When you configure IPv4 and IPv6 source guard together on an interface, it is recommended to use **ip verify source mac-check** instead of **ip verify source** . IPv4 connectivity on a given port might break due to two different filtering rules set — one for IPv4 (IP-filter) and the other for IPv6 (IP-MAC filter).

- You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.
- PVLAN and Source/Prefix Guard cannot be applied together.

For more information on IPv6 Source Guard, see the IPv6 Source Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Prefix Guard—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

  For more information on IPv6 Prefix Guard, see the IPv6 Prefix Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Destination Guard—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.

  For more information about IPv6 Destination Guard, see the IPv6 Destination Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Neighbor Discovery Multicast Suppress—The IPv6 Neighbor Discovery multicast suppress feature is an IPv6 snooping feature that runs on a switch or a wireless controller and is used to reduce the amount of control traffic necessary for proper link operations.

- DHCPv6 Relay—Lightweight DHCPv6 Relay Agent—The DHCPv6 Relay—Lightweight DHCPv6 Relay Agent feature allows relay agent information to be inserted by an access node that performs a link-layer bridging (non-routing) function. Lightweight DHCPv6 Relay Agent (LDRA) functionality can be implemented in existing access nodes, such as DSL access multiplexers (DSLAMs) and Ethernet switches, that do not support IPv6 control or routing functions. LDRA is used to insert relay-agent options in DHCP version 6 (DHCPv6) message exchanges primarily to identify client-facing interfaces. LDRA functionality can be enabled on an interface and on a VLAN.

  Note If an LDRA device is directly connected to a client, the interface must have the pool configuration to fetch the specific subnet or link information at the server side. In this case, if the LDRA device is present in different subnets or links, the server may not be able to fetch the correct subnet. You can now configure the pool name in the interface so as to choose the proper subnet or link for the client.

  For more information about DHCPv6 Relay, See the DHCPv6 Relay—Lightweight DHCPv6 Relay Agent section of the IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG.

How to configure an IPv6 Snooping Policy

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **IPv6 snooping policy** *policy -name*
4. [**data-glean** | **default** | **device-role** [**node** | **switch**] | **limit** {**address-count***value* } | **no** | **protocol** [**all** | **nodhcp** | **ndp**] | **security-level** [**glean** | **guard** | **inspect**] | **tracking** [**disable** | **enable**] | **trusted-port** }
5. **exit**
6. **show ipv6 snooping policy***policy-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	**IPv6 snooping policy** *policy -name*	Creates a snooping policy in global configuration mode.
Step 4	[**data-glean**	**default**	**device-role** [**node**	**switch**]	**limit** {**address-count***value* }	**no**	**protocol** [**all**	**nodhcp**	**ndp**]	**security-level** [**glean**	**guard**	**inspect**]	**tracking** [**disable**	**enable**]	**trusted-port** }	Enables data address gleaning, validates messages against various criteria, specifies the security level for messages. • (Optional) **data-glean**—Enables data address gleaning. This option is disabled by default. • (Optional) **default**—Sets all default options. • (Optional) **device-role** [**node**	**switch**]—Qualifies the role of the device attached to the port. • (Optional) **limit** {**address-count** *value*}—Limits the number ofaddresses allowed per target. • (Optional) **no**—Negates a command or set its defaults. • (Optional) **protocol** [**all**	**dhcp**	**ndp**]—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is **all**. To change the default, use the **no protocol** command. • (Optional) **security-level** [**glean**	**guard**	**inspect**]—Specifies the level of security enforced by the feature. • **glean**—Gleans addresses from messages and populates the binding table without any verification. • **guard**—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option.

	Command or Action	Purpose
		• **inspect**—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) **tracking** [**disable** | **enable**]—Overrides the default tracking behavior and specifies a tracking option. • (Optional) **trusted-port**—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is also given preference in case of a collision while making an entry in the table.
Step 5	**exit**	Exits the snooping policy configuration mode.
Step 6	**show ipv6 snooping policy***policy-name*	Displays the snooping policy configuration.

How to Attach an IPv6 Snooping Policy to an Interface or VLAN

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
   - **interface** *type number*
   - **switchport**
   - **ipv6 snooping** [**attach-policy** *policy_name*]

   OR

   - **vlan configuration** *vlan list*
   - **ipv6 snooping attach-policy** *policy-name*
4. **show ipv6 snooping policy** *policy-name*
5. **show ipv6 neighbors binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	Perform one of the following tasks: • **interface** *type number* • **switchport** • **ipv6 snooping** [**attach-policy** *policy_name*] OR • **vlan configuration** *vlan list* • **ipv6 snooping attach-policy** *policy-name*	Specifies an interface type and number, and enters the interface configuration mode. **Note** type can be physical interface or ether-channel. Configures the interface as a Layer 2 port. Attaches the snooping policy (where data gleaning is enabled) to an interface. Specifies the port and the policy that is attached to the port. **Note** If you have enabled **data-glean** on a snooping policy, you must attach it to an interface and not a VLAN.
Step 4	**show ipv6 snooping policy** *policy-name*	Displays the snooping policy configuration.
Step 5	**show ipv6 neighbors binding**	Displays the binding table entries populated by the snooping policy.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on a Device

To attach an IPV6 Neighbor Discovery Multicast Suppress policy on a device, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 nd suppress policy** *policy-name*
4. **mode dad-proxy**
5. **mode full-proxy**
6. **mode mc-proxy**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	**ipv6 nd suppress policy** *policy-name*	Defines the Neighbor Discovery suppress policy name and enters Neighbor Discovery suppress policy configuration mode.
Step 4	**mode dad-proxy**	Enables Neighbor Discovery suppress in IPv6 DAD proxy mode.

	Command or Action	Purpose
Step 5	**mode full-proxy**	Enables Neighbor Discovery suppress to proxy multicast and unicast Neighbor Solicitation messages.
Step 6	**mode mc-proxy**	Enables Neighbor Discovery suppress to proxy multicast Neighbor Solicitation messages.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on an Interface

To attach an IPv6 Neighbor Discovery Multicast Suppress policy on an interface, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
   - **interface** *type number*
   - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]

     OR

   - **vlan configuration** *vlan-id*
   - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	Perform one of the following tasks: • **interface** *type number* • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add**	**except**	**none**	**remove**	**all**} *vlan* [*vlan1, vlan2, vlan3...*]]] OR • **vlan configuration** *vlan-id*	Specifies an interface type and number, and places the device in interface configuration mode. Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.

	Command or Action	Purpose
	• **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]	
Step 4	**exit**	Exists the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy to a Layer 2 EtherChannel Interface

To attach an IPv6 Neighbor Discovery Multicast Suppress policy on an EtherChannel interface, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
   - **interface port-channel** *port-channel-number*
   - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** \| **except** \| **none** \| **remove** \| **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]

     OR

   - **vlan configuration** *vlan-id*
   - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** \| **except** \| **none** \| **remove** \| **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	Perform one of the following tasks: • **interface port-channel** *port-channel-number* • **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]] OR • **vlan configuration** *vlan-id*	Specifies an interface type and port number and places the switch in the port channel configuration mode. Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.

	Command or Action	Purpose
	• **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]	
Step 4	**exit**	Exists the interface configuration mode.

How to Configure an IPv6 DHCP Guard Policy

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 dhcp guard policy** *policy-name*
4. [**default** | **device-role** [**client** | **server**] |**no** | **exit** | **trusted-port**]
5. **exit**
6. Perform one of the following tasks:
   - **interface** *type number*
   - **ipv6 dhcp guard attach-policy** *policy-name*

   OR

   - **vlan configuration** *vlan-id*
   - **ipv6 dhcp guard attach-policy** *policy-name*
7. **show ipv6 dhcp guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	**ipv6 dhcp guard policy** *policy-name*	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.
Step 4	[**default** | **device-role** [**client** | **server**] |**no** | **exit** | **trusted-port**]	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is **client**. • **client**—Default value, specifies that the attached device is a client. Server messages are dropped on this port.

	Command or Action	Purpose
		• **server**—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port. (Optional) **trusted-port**—Sets the port to a trusted mode. No further policing takes place on the port. **Note** If you configure a trusted port then the device-role option is not available.
Step 5	**exit**	Exits the DHCP guard policy global configuration mode.
Step 6	Perform one of the following tasks: • **interface** *type number* • **ipv6 dhcp guard attach-policy** *policy-name* OR • **vlan configuration** *vlan-id* • **ipv6 dhcp guard attach-policy** *policy-name*	Specifies an interface type and number and enters the interface configuration mode. Attaches the DHCP guard policy to an interface or VLAN.
Step 7	**show ipv6 dhcp guard policy** *policy_name*	Displays the DHCP guard policy configuration.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan configuration 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

How to Configure IPv6 Source Guard

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 source-guard policy** *policy_name*

4. [**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]
5. **ipv6 source-guard** [**attach-policy***policy-name*]
6. **exit**
7. **show ipv6 source-guard policy***policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	**ipv6 source-guard policy** *policy_name*	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]	Defines the IPv6 Source Guard policy. • **deny global-autoconf**—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and the administrator wants to block hosts with self-configured addresses to send traffic. • **permit link-local**—Allows all data traffic that is sourced by a link-local address.
Step 5	**ipv6 source-guard** [**attach-policy***policy-name*]	Specifies the policy name. (Optional) **attach-policy** *policy-name*—Filters based on the policy name
Step 6	**exit**	Exits the source guard policy configuration mode.
Step 7	**show ipv6 source-guard policy***policy_name*	Shows the policy configuration and all the interfaces where the policy is applied.

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.

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.

To configure a DRP for a router on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode and identifies the Layer 3 interface on which you want to specify the DRP.
Step 4	**ipv6 nd router-preference** {**high** | **medium** | **low**} **Example:** `Switch(config-if)# ipv6 nd router-preference medium`	Specifies a DRP for the router on the switch interface.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show ipv6 interface** **Example:** `Switch# show ipv6 interface`	Verifies the configuration.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring Default Router Preference: Example, on page 52

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.

To change the ICMP rate-limiting parameters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 icmp error-interval** *interval* [*bucketsize*] **Example:** `Switch(config)# ipv6 icmp error-interval 50 20`	Configures 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 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ipv6 interface** [*interface-id*] **Example:** `Switch# show ipv6 interface gigabitethernet0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring IPv6 ICMP Rate Limiting: Example, on page 53

Configuring Cisco Express Forwarding and distributed Cisco Express Forwarding for IPv6

Cisco Express Forwarding is a Layer 3 IP switching technology to improve network performance. Cisco Express Forwarding implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. It is less CPU-intensive than fast-switching route-caching, allowing more CPU processing power to be dedicated to packet forwarding. IPv4 Cisco Express Forwarding and distributed Cisco Express Forwarding are enabled by default. IPv6 Cisco Express Forwarding and distributed Cisco Express Forwarding are disabled by default, but automatically enabled when you configure IPv6 routing.

IPv6 Cisco Express Forwarding and distributed Cisco Express Forwarding are automatically disabled when IPv6 routing is unconfigured. IPv6 Cisco Express Forwarding and distributed Cisco Express Forwarding cannot disabled through configuration. You can verify the IPv6 state by entering the **show ipv6 cef** privileged EXEC command.

To route IPv6 unicast packets, you must first globally configure forwarding of IPv6 unicast packets by using the **ipv6 unicast-routing** global configuration command, and you must configure an IPv6 address and IPv6 processing on an interface by using the **ipv6 address** interface configuration command.

For more information about configuring Cisco Express Forwarding and distributed Cisco Express Forwarding, see *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring Static Routing for IPv6

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.

To configure static IPv6 routing, perform this procedure:

Before you begin

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.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Switch# configure terminal`	
Step 3	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*] **Example:** `Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130`	Configures 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*—Specifies 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 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose

Step 5	Use one of the following:
	• **show ipv6 static** [*ipv6-address* | *ipv6-prefix/prefix length*] [**interface** *interface-id*] [**detail**]][**recursive**] [**detail**]
	• **show ipv6 route static** [*updated*]
	Example:
	`Switch# show ipv6 static 2001:0DB8::/32 interface gigabitethernet2/0/1`
	or
	`Switch# show ipv6 route static`
		Verifies your entries by displaying the contents of the IPv6 routing table.
		• **interface** *interface-id*—(Optional) Displays only those static routes with the specified interface as an egress interface.
		• **recursive**—(Optional) Displays 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) Displays 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 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring Static Routing for IPv6: Example, on page 54

Configuring RIP for IPv6

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

To configure RIP routing for IPv6, perform this procedure:

Before you begin

Before configuring the switch to run IPv6 RIP, 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 any Layer 3 interfaces on which IPv6 RIP is to be enabled.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	`Switch> enable`	
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router rip** *name* **Example:** `Switch(config)# ipv6 router rip cisco`	Configures an IPv6 RIP routing process, and enters router configuration mode for the process.
Step 4	**maximum-paths** *number-paths* **Example:** `Switch(config-router)# maximum-paths 6`	(Optional) Define the maximum number of equal-cost routes that IPv6 RIP can support. The range is from 1 to 32, and the default is 16 routes.
Step 5	**exit** **Example:** `Switch(config-router)# exit`	Returns to global configuration mode.
Step 6	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 7	**ipv6 rip** *name* **enable** **Example:** `Switch(config-if)# ipv6 rip cisco enable`	Enables the specified IPv6 RIP routing process on the interface.
Step 8	**ipv6 rip** *name* **default-information** {**only** | **originate**} **Example:** `Switch(config-if)# ipv6 rip cisco default-information only`	(Optional) Originates the IPv6 default route (::/0) into the RIP routing process updates sent from the specified interface. **Note** To avoid routing loops after the IPv6 default route (::/0) is originated from any interface, the routing process ignores all default routes received on any interface.

	Command or Action	Purpose
		• **only**—Select to originate the default route, but suppress all other routes in the updates sent on this interface. • **originate**—Select to originate the default route in addition to all other routes in the updates sent on this interface.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Step 10	Use one of the following:
	• **show ipv6 rip** [*name*] [**interface** *interface-id*] [**database**] [**next-hops**]
	• **show ipv6 rip**
	Example:
	`Switch# show ipv6 rip cisco interface gigabitethernet 2/0/1`
	or
	`Switch# show ipv6 rip`
		• Displays information about current IPv6 RIP processes.
		• Displays the current contents of the IPv6 routing table.

Step 11	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring RIP for IPv6: Example, on page 54

Configuring OSPF for IPv6

For more information about configuring OSPF routing for IPv6, see the “Implementing OSPF for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

To configure OSPF routing for IPv6, perform this procedure:

Before you begin

You can customize OSPF for IPv6 for your network. However, the defaults for OSPF in IPv6 are set to meet the requirements of most customers and features.

Follow these guidelines:

- Be careful when changing the defaults for IPv6 commands. Changing the defaults might adversely affect OSPF for the IPv6 network.
- Before you enable IPv6 OSPF on an interface, 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 Layer 3 interfaces on which you are enabling IPv6 OSPF.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router ospf** *process-id* **Example:** `Switch(config)# ipv6 router ospf 21`	Enables OSPF router configuration mode for the process. The process ID is the number assigned administratively when enabling the OSPF for IPv6 routing process. It is locally assigned and can be a positive integer from 1 to 65535.
Step 4	**area** *area-id* **range** {*ipv6-prefix/prefix length*} [**advertise** | **not-advertise**] [**cost** *cost*] **Example:** `Switch(config)# area .3 range 2001:0DB8::/32 not-advertise`	(Optional) Consolidates and summarizes routes at an area boundary. • *area-id*—Identifier of the area about which routes are to be summarized. It can be specified as either a decimal value or as an IPv6 prefix. • *ipv6-prefix/prefix length*—The destination IPv6 network and 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. • **advertise**—(Optional) Sets the address range status to advertise and generate a Type 3 summary link-state advertisement (LSA). • **not-advertise**—(Optional) Sets the address range status to DoNotAdvertise. The Type 3 summary LSA is suppressed, and component networks remain hidden from other networks. • **cost** *cost*—(Optional) Sets the metric or cost for this summary route, which is used during OSPF SPF

	Command or Action	Purpose
		calculation to determine the shortest paths to the destination. The value can be 0 to 16777215.
Step 5	**maximum paths** *number-paths* **Example:** `Switch(config)# maximum paths 16`	(Optional) Defines the maximum number of equal-cost routes to the same destination that IPv6 OSPF should enter in the routing table. The range is from 1 to 32, and the default is 16 paths.
Step 6	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 7	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 8	**ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*] **Example:** `Switch(config-if)# ipv6 ospf 21 area .3`	Enables OSPF for IPv6 on the interface. • **instance** *instance-id*—(Optional) Instance identifier.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Step 10	Use one of the following:
	• **show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-id*]
	• **show ipv6 ospf** [*process-id*] [*area-id*]
	Example:
	`Switch# show ipv6 ospf 21 interface gigabitethernet2/0/1`
	or
	`Switch# show ipv6 ospf 21`
		• Displays information about OSPF interfaces.
		• Displays general information about OSPF routing processes.

Step 11	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	`Switch# copy running-config startup-config`	

Tuning LSA and SPF Timers for OSPFv3 Fast Convergence

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf***process-id*
4. **timers lsa arrival** *milliseconds*
5. **timers pacing flood***milliseconds*
6. **timers pacing lsa-group***seconds*
7. **timers pacing retransmission***milliseconds*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router ospf***process-id*	Enables OSPFv3 router configuration mode.
Step 4	**timers lsa arrival** *milliseconds*	Sets the minimum interval at which the software accepts the same LSA from OSPFv3 neighbors.
Step 5	**timers pacing flood***milliseconds*	Configures LSA flood packet pacing.
Step 6	**timers pacing lsa-group***seconds*	Changes the interval at which OSPFv3 LSAs are collected into a group and refreshed, checksummed, or aged.
Step 7	**timers pacing retransmission***milliseconds*	Configures LSA retransmission packet pacing in OSPFv3.
Step 8	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	`Switch(config-if)# end`	

Configuring LSA and SPF Throttling for OSPFv3 Fast Convergence

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf***process-id*
4. **timers throttle spf** *spf-start spf-hold spf-max-wait*
5. **timers throttle lsa***start-intervalhold-intervalmax-interval*
6. **timers lsa arrival***milliseconds*
7. **timers pacing flood***milliseconds*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router ospf***process-id*	Enables OSPFv3 router configuration mode.
Step 4	**timers throttle spf** *spf-start spf-hold spf-max-wait*	Turns on SPF throttling.
Step 5	**timers throttle lsa***start-intervalhold-intervalmax-interval*	Sets rate-limiting values for OSPFv3 LSA generation.
Step 6	**timers lsa arrival***milliseconds*	Sets the minimum interval at which the software accepts the same LSA from OSPFv3 neighbors.
Step 7	**timers pacing flood***milliseconds*	Configures LSA flood packet pacing.
Step 8	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring EIGRP for IPv6

Before configuring the switch to run IPv6 EIGRP, enable routing by entering the **ip routing global configuration** command, enable the forwarding of IPv6 packets by entering the **ipv6 unicast-routing global** configuration command, and enable IPv6 on any Layer 3 interfaces on which you want to enable IPv6 EIGRP.

To set an explicit router ID, use the **show ipv6 eigrp** command to see the configured router IDs, and then use the **router-id** command.

As with EIGRP IPv4, you can use EIGRPv6 to specify your EIGRP IPv6 interfaces and to select a subset of those as passive interfaces. Use the **passive-interface** command to make an interface passive, and then use the **no passive-interface** command on selected interfaces to make them active. EIGRP IPv6 does not need to be configured on a passive interface.

For more configuration procedures, see the "Implementing EIGRP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring HSRP for IPv6

Hot Standby Router Protocol (HSRP) for IPv6 provides routing redundancy for routing IPv6 traffic not dependent on the availability of any single router.

When HSRP for IPv6 is enabled on a switch, IPv6 hosts learn of available IPv6 routers through IPv6 neighbor discovery router advertisement messages. An HSRP IPv6 group has a virtual MAC address that is derived from the HSRP group number. The group has a virtual IPv6 link-local address that is, by default, derived from the HSRP virtual MAC address. Periodic messages are sent for the HSRP virtual IPv6 link-local address when the HSRP group is active.

When configuring HSRP for IPv6, you must enable HSRP version 2 (HSRPv2) on the interface.

Note Before configuring an HSRP for IPv6 group, you must enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command and enable IPv6 on the interface on which you will configure an HSRP for IPv6 group.

Enabling HSRP Version 2

For more information about configuring HSRP for IPv6, see the "Configuring First Hop Redundancy Protocols in IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and enters the Layer 3 interface on which you want to specify the standby version.
Step 3	**standby version** {**1** | **2**} **Example:** `Switch(config-if)# standby version 2`	Sets the HSRP version. Enter **2** to change the HSRP version. The default is 1.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show standby** **Example:** `Switch# show standby`	Verifies the configuration.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling an HSRP Group for IPv6

This task explains how to create or enable HSRP for IPv6 on a Layer 3 interface.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:**	Enters interface configuration mode, and enters the Layer 3 interface on which you want to enable HSRP for IPv6.

	Command or Action	Purpose
	`Switch(config)# interface gigabitethernet 1/0/1`	
Step 3	**standby** [*group-number*] **ipv6** {*link-local-address* | **autoconfig**} **Example:** `Switch(config-if)# standby 2 ipv6 auto config`	Creates (or enables) the HSRP for IPv6 group. • (Optional) *group-number*—The group number on the interface for which HSRP is being enabled. The range is 0 to 4095. The default is 0. If there is only one HSRP group, you do not need to enter a group number. • Enter the link-local address of the Hot Standby router interface, or enable the link-local address to be generated automatically from the link-local prefix and a modified EUI-64 format interface identifier, where the EUI-64 interface identifier is created from the relevant HSRP virtual MAC address.
Step 4	**standby** [*group-number*] **preempt** [**delay** {**minimum** *seconds* | **reload** *seconds* | **sync** *seconds*}] **Example:** `Switch(config-if)# standby 2 preempt delay reload 0`	Configures the router to **preempt**, which means that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **delay**—Sets to cause the local router to postpone taking over the active role for the shown number of seconds. The range is 0 to 3600 (1 hour). The default is 0 (no delay before taking over). • (Optional) **reload**—Sets the preemption delay, in seconds, after a reload. The delay period applies only to the first interface-up event after the router reloads. • (Optional) **sync**—Sets the maximum synchronization period, in seconds, for IP redundancy clients. Use the **no** form of the command to restore the default values.
Step 5	**standby** [*group-number*] **priority** *priority* **Example:** `Switch(config-if)# standby 2 priority 200`	Sets a **priority** value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. Use the **no** form of the command to restore the default values.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	**show standby** [*interface-id* [*group-number*]] **Example:** `Switch# show standby gigabitethernet 1/0/1 2`	Verifies the configuration.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Enabling an HSRP Group for IPv6: Example, on page 52

Configuring Multi-VRF CE

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE) when the it is running the IP services or advanced IP Services feature set. Multi-VRF CE allows a service provider to support two or more VPNs with overlapping IP addresses.

Note The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs.

IPv6 multicast routing is not supported on a VRF associated interface.

Default Multi-VRF CE Configuration

Table 2: Default VRF Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
Forwarding table	The default for an interface is the global routing table.

Configuring VRFs

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference*.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 unicast-routing** **Example:** `Switch(config)# ipv6 unicast routing`	Enables IPv6 unicast routing.
Step 3	**vrf definition** *vrf-name* **Example:** `Switch(config)# vrf definition vpn1`	Names the VRF, and enters VRF configuration mode.
Step 4	**address family** *ipv6* **Example:** `Switch(config)# address family ipv6`	Specifies the IPv6 address family and enter address family configuration mode.
Step 5	**rd** *route-distinguisher* **Example:** `Switch(config-vrf)# rd 100:2`	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y)
Step 6	**route-target** {**export** | **import** | **both**} *route-target-ext-community* **Example:** `Switch(config-vrf)# route-target both 100:2`	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The *route-target-ext-community* should be the same as the *route-distinguisher* entered in Step 4.
Step 7	**import map** *route-map* **Example:** `Switch(config-vrf)# import map importmap1`	(Optional) Associates a route map with the VRF.
Step 8	**interface** *interface-id* **Example:** `Switch(config-vrf)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or SVI.
Step 9	**vrf forwarding** *vrf-name* **Example:**	Associates the VRF with the Layer 3 interface.

	Command or Action	Purpose
	`Switch(config-if)# vrf forwarding vpn1`	
Step 10	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 11	**show vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*] **Example:** `Switch# show vrf interfaces vpn1`	Verifies the configuration. Displays information about the configured VRFs.
Step 12	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring VRF-Aware Services

These services are VRF-Aware:

- ARP
- Ping
- Simple Network Management Protocol (SNMP)
- Hot Standby Router Protocol (HSRP)
- Unicast Reverse Path Forwarding (uRPF)
- Syslog
- Traceroute
- FTP and TFTP

Note The switch does not support VRF-aware services for Unicast Reverse Path Forwarding (uRPF) or Network Time Protocol (NTP).

Configuring VRF-Aware Services for Neighbor Discovery

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

Procedure

	Command or Action	Purpose
Step 1	**show ipv6 neighbors vrf***vrf-name* **Example:** `Switch# show ipv6 neighbors vrf vpn1`	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for PING

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release* .

Procedure

	Command or Action	Purpose
Step 1	**ping vrf***vrf-name***ipv6***ipv6-address* **Example:** `Switch# ping vrf vpn1 ipv6`	Displays the ARP table in the specified VRF.

Configuring VRF-Aware Services for HSRP

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.4*.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch# interface gigabitethernet1/0/1`	Enters interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	**no switchport** **Example:** `Switch# no switchport`	Removes the interface from Layer 2 configuration mode if it is a physical interface.
Step 4	**vrf forwarding***vrf-name* **Example:**	Configures VRF on the interface.

	Command or Action	Purpose
	`Switch# vrf forwarding vpn1`	
Step 5	**ipv6 address***ipv6 address* **Example:** `Switch# ipv6 address 2001::DB8:1/64`	Enters the IPv6 address for the interface.
Step 6	**standby 1 ipv6***ipv6 address* **Example:** `Switch# standby 1 ipv6 2001::DB8:1/64`	Enables HSRP and configures the virtual IP address.
Step 7	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Configuring VRF-Aware Services for Traceroute

For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release* .

Procedure

	Command or Action	Purpose
Step 1	**traceroute vrf** *vrf-nameipv6-address* **Example:** `Switch# traceroute vrf`**`vpn1 2001::DB8:1/64`**	Specifies the name of a VPN VRF in which to find the destination address.

Configuring VRF-Aware Services for FTP and TFTP

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip ftp source-interface** *interface-type interface-number* **Example:**	Specifies the source IP address for FTP connections.

	Command or Action	Purpose
	`Switch(config)# ip ftp source-interface gigabitethernet 1/0/2`	
Step 3	**end** **Example:** `Switch(config)#end`	Returns to privileged EXEC mode.
Step 4	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 5	**ip tftp source-interface** *interface-type interface-number* **Example:** `Switch(config)# ip tftp source-interface gigabitethernet 1/0/2`	Specifies the source IP address for TFTP connections.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**end** **Example:** `Switch(config)#end`	Returns to privileged EXEC mode.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.

Note To configure an EIGRP routing process to run within a VRF instance, you must configure an autonomous-system number by entering the **autonomous-system** *autonomous-system-number* address-family configuration mode command.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Switch# configure terminal`	
Step 2	**router ospfv3** *process-id* **Example:** `Switch(config)# router ospfv3 1`	Enables OSPF routing, specifies a VPN forwarding table, and enter router configuration mode.
Step 3	**router** *router-id* **Example:** `Switch(config)# router router-id`	Specifies the OSPF router-id in IP address format for this OSPFv3 process.
Step 4	**log-adjacency-changes** **Example:** `Switch(config-router)# log-adjacency-changes`	(Optional) Logs changes in the adjacency state. This is the default state.
Step 5	**address-family ipv6 unicast vrf** *vrf-name* **Example:** `Switch(config-router)# address-family ipv6 unicast vrf vpn1`	Enters address family command mode for the VRF.
Step 6	**area** *area-id normal* **Example:** `Switch(config-router)# area 2`	Specifies OSPFv3 area parameters and type.
Step 7	**redistribute bgp** *autonomous-system-number* **Example:** `Switch(config-router)# redistribute bgp 10`	Redistributes routes from BGP routing process to OSPF routing process.
Step 8	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 9	**show ospfv3 vrf** *vrf-name* **Example:** `Switch# show ospfv3 vrf vpn1`	Verifies the configuration of the OSPFv3 network.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring BGP PE to CE Routing Sessions

Procedure

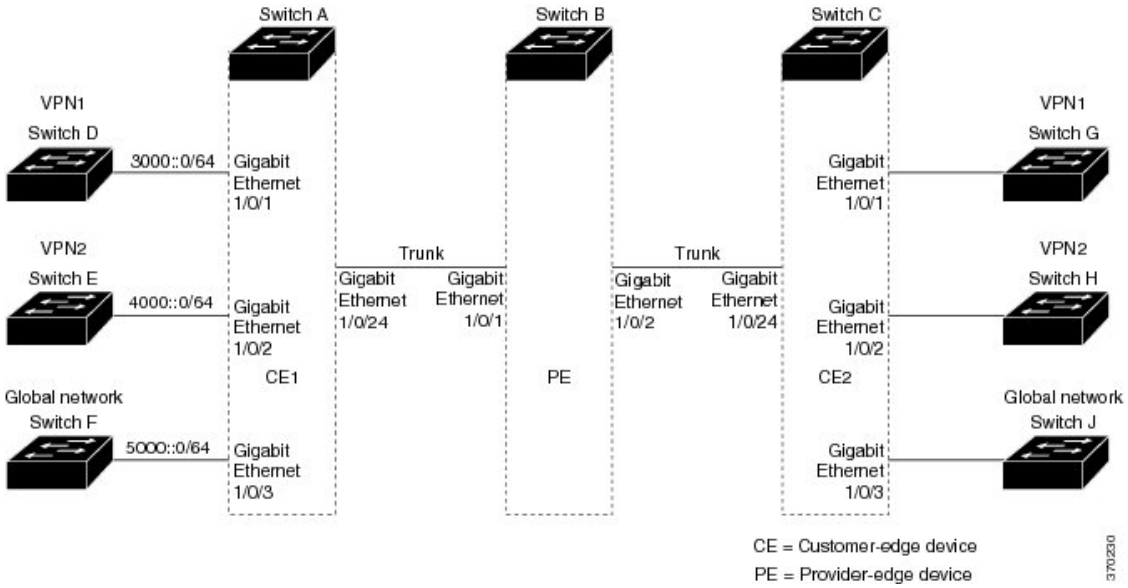
	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system-number* **Example:** `Switch(config)# router bgp 2`	Configures the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
Step 3	**bgp router id** *router-id* **Example:** `Switch(config)# bgp router-id`	Configures a fixed 32-bit router id as the identifier of the local router running BGP.
Step 4	**redistribute ospf** *process-id* **Example:** `Switch(config-router)# redistribute ospf 1`	Sets the switch to redistribute OSPF internal routes.
Step 5	**address-family ipv6 vrf** *vrf-name* **Example:** `Switch(config-router)# address-family ipv6 vrf vpn1`	Defines BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
Step 6	**network** *ipv6 network-number* **Example:** `Switch(config-router)# network ipv6 255.255.255.0`	Specifies an IPv6 Network number to announce via BGP.
Step 7	**neighbor** *ipv6 address* **remote-as** *as-number* **Example:** `Switch(config-router)# neighbor 10.1.1.2 remote-as 2`	Defines a BGP session between PE and CE routers.
Step 8	**neighbor** *address* **activate** **Example:** `Switch(config-router)# neighbor 10.2.1.1 activate`	Activates the advertisement of the IPv4 address family.

	Command or Action	Purpose
Step 9	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 10	**show bgp vrf** *vrf-name* **Example:** `Switch# show ip bgp ipv4 neighbors`	Verifies BGP configuration on the VRF.
Step 11	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Multi-VRF CE Configuration Example

OSPF is the protocol used in VPN1, VPN2, and the global network. BGP is used in the CE to PE connections. The examples following the illustration show how to configure a switch as CE Switch A, and the VRF configuration for customer switches D and E. Commands for configuring CE Switch C and the other customer switches are not included but would be similar.

Figure 1: Multi-VRF CE Configuration Example



On Switch A, enable routing and configure VRF.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ipv6 unicast-routing
Switch(config)# vrf definition v11
```

```
Switch(config-vrf)# rd 11:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf)# exit
Switch(config-vrf)# vrf definition v12
Switch(config-vrf)# rd 12:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# end
```

Configure the physical interfaces on Switch A. Gigabit Ethernet interface 1/0/24 is a trunk connection to the PE. Gigabit Ethernet ports 1/0/1 and 1/0/2 connect to VPNs.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface GigabitEthernet 1/0/1
Switch(config-if)# switchport access vlan 208
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface gigabitEthernet 1/0/2
Switch(config-if)# switchport access vlan 118
Switch(config-if)# no ip address
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet 1/0/24
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# exit
```

Configure the VLANs used on Switch A. VLAN 10 is used by VRF 11 between the CE and the PE. VLAN 20 is used by VRF 12 between the CE and the PE. VLANs 118 and 208 are used for the VPNs that include Switch E and Switch D, respectively:

```
Switch(config)# interface vlan10
Switch(config-if)# vrf forwarding v11
Switch(config-if)# ipv6 address 1000::1/64
Switch(config-if)# exit

Switch(config)# interface vlan20
Switch(config-if)# vrf forwarding v12
Switch(config-if)# ipv6 address 2000::1/64
Switch(config-if)# exit

Switch(config)# interface vlan208
Switch(config-if)# vrf forwarding v11
Switch(config-if)# ipv6 address 3000::1/64
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# vrf forwarding v12
Switch(config-if)# ipv6 address 4000::1/64
Switch(config-if)# exit
```

Configure OSPFv3 routing on VPN1 and VPN2.

```
Switch(config)# router ospfv3 1
Switch(config-router)# router-id 1.1.1.1
Switch(config-router)# address-family ipv6 unicast vrf v11
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute bgp 800
Switch(config-router)# exit
Switch(config)# router ospfv3 2
Switch(config-router)# router-id 2.2.2.2
```

```
Switch(config-router)# address-family ipv6 unicast vrf v12
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute bgp 800
Switch(config-router-af)# exit
Switch(config-router)# exit
Switch(config)# exit
```

Configure BGP for CE to PE routing.

```
Switch(config)# router bgp 800
Switch(config-router)# bgp router-id 8.8.8.8
Switch(config-router)# address-family ipv6 vrf v11
Switch(config-router-af)# redistribute ospf 1
Switch(config-router-af)# neighbor 1000::2 remote-as 100
Switch(config-router-af)# neighbor 1000::2 activate
Switch(config-router-af)# network 3000::/64
Switch(config-router-af)# exit

Switch(config)# address-family ipv6 vrf v12
Switch(config-router-af)# redistribute ospf 2
Switch(config-router-af)# neighbor 2000::2 remote-as 100
Switch(config-router-af)# neighbor 2000::2 activate
Switch(config-router-af)# network 4000::/64
```

Switch D belongs to VPN 1. Configure the connection to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ipv6 unicast-routing
Switch(config)# interface GigabitEthernet 5/0/16
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 3000::2/64
Switch(config-if)# exit

Switch(config-router)# router ospfv3 101
Switch(config-router)# address-family ipv6
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute connected
Switch(config-router-af)# exit
Switch(config-router)# exit
```

Switch E belongs to VPN 2. Configure the connection to Switch A by using these commands.

```
Switch(config)# ipv6 unicast-routing
Switch(config)# interface GigabitEthernet 3/0/13
Switch(config-if)# switchport access vlan 20
Switch(config-if)# exit
Switch(config)# interface vlan 20
Switch(config-if)# ipv6 address 4000::2/64


Switch(config)# router ospfv3 101
Switch(config-router)# address-family ipv6
Switch(config-router-af)# area 0 normal
Switch(config-router-af)# redistribute connected
Switch(config-router-af)# end
```

When used on switch B (the PE router), these commands configure only the connections to the CE device, Switch A.

```
Switch(config)# vrf definition v1
Switch(config-vrf)# rd 1:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# exit
Switch(config-vrf)# exit

Switch(config)# vrf definition v2
Switch(config-vrf)# rd 2:1
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# exit
Switch(config-vrf)# exit

Switch(config-if)# interface g 1/0/2
Switch(config-if)# vrf forwarding v1
Switch(config-if)# ipv6 address 1000::2/64
Switch(config-if)# exit
Switch(config)# interface g 1/0/4
Switch(config-if)# vrf forwarding v2
Switch(config-if)# ipv6 address 2000::2/64

Switch(config-if)# interface gigabitEthernet 1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk

Switch(config)# router bgp 100
Switch(config-router)# address-family ipv6 vrf v1
Switch(config-router-af)# neighbor 1000::1 remote-as 100
Switch(config-router-af)# neighbor 1000::1 activate
Switch(config-router-af)# network 3000::/64
Switch(config-router-af)# exit
Switch(config-router)# address-family ipv6 vrf v2
Switch(config-router-af)# neighbor 2000::1 remote-as 100
Switch(config-router-af)# neighbor 2000::1 activate
Switch(config-router-af)# network 4000::/64
```

Displaying Multi-VRF CE Status

Table 3: Commands for Displaying Multi-VRF CE Information

Command	Purpose
show ipv6 protocols vrf*vrf-name*	Displays routing protocol information associated with a VRF.
show ipv6 route vrf*vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	Displays IP routing table information associated with a VRF.
show ipv6 vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	Displays information about the defined VRF instances.

Displaying IPv6

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

Table 4: Command for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Displays a summary of access lists.
show ipv6 cef	Displays Cisco Express Forwarding for IPv6.
show ipv6 interface *interface-id*	Displays IPv6 interface status and configuration.
show ipv6 mtu	Displays IPv6 MTU per destination cache.
show ipv6 neighbors	Displays IPv6 neighbor cache entries.
show ipv6 prefix-list	Displays a list of IPv6 prefix lists.
show ipv6 protocols	Displays a list of IPv6 routing protocols on the switch.
show ipv6 rip	Displays IPv6 RIP routing protocol status.
show ipv6 route	Displays IPv6 route table entries.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.

Related Topics

Displaying IPv6: Example, on page 54

Configuring DHCP for IPv6 Address Assignment

This section describes only the DHCPv6 address assignment. For more information about configuring the DHCPv6 client, server, or relay agent functions, see the “Implementing DHCP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Default DHCPv6 Address Assignment Configuration

By default, no DHCPv6 features are configured on the switch.

DHCPv6 Address Assignment Configuration Guidelines

When configuring DHCPv6 address assignment, consider these guidelines:

- In the procedures, the specified interface must be one of these Layer 3 interfaces:
  - DHCPv6 IPv6 routing must be enabled on a Layer 3 interface.
  - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* command.
  - EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel port-channel-number** command.

- The switch can act as a DHCPv6 client, server, or relay agent. The DHCPv6 client, server, and relay function are mutually exclusive on an interface.

Enabling DHCPv6 Server Function (CLI)

Use the **no** form of the DHCP pool configuration mode commands to change the DHCPv6 pool characteristics. To disable the DHCPv6 server function on an interface, use the **no ipv6 dhcp server** interface configuration command.

To enable the DHCPv6 server function on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 dhcp pool** *poolname* **Example:** `Switch(config)# ipv6 dhcp pool 7`	Enters DHCP pool configuration mode, and define the name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0).
Step 4	**address prefix** *IPv6-prefix* {**lifetime**} {*t1 t1* | **infinite**} **Example:** `Switch(config-dhcpv6)# address prefix 2001:1000::0/64 lifetime 3600`	(Optional) Specifies an address prefix for address assignment. This address must be in hexadecimal, using 16-bit values between colons. **lifetime** *t1 t1*—Specifies a time interval (in seconds) that an IPv6 address prefix remains in the valid state. The range is 5 to 4294967295 seconds. Specify **infinite** for no time interval.
Step 5	**link-address** *IPv6-prefix* **Example:** `Switch(config-dhcpv6)# link-address 2001:1002::0/64`	(Optional) Specifies a link-address IPv6 prefix. When an address on the incoming interface or a link-address in the packet matches the specified IPv6 prefix, the server uses the configuration information pool. This address must be in hexadecimal, using 16-bit values between colons.

	Command or Action	Purpose
Step 6	**vendor-specific** *vendor-id* **Example:** `Switch(config-dhcpv6)# vendor-specific 9`	(Optional) Enters vendor-specific configuration mode and specifies a vendor-specific identification number. This number is the vendor IANA Private Enterprise Number. The range is 1 to 4294967295.
Step 7	**suboption** *number* {**address** *IPv6-address* | **ascii** *ASCII-string* | **hex** *hex-string*} **Example:** `Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::`	(Optional) Enters a vendor-specific suboption number. The range is 1 to 65535. Enter an IPv6 address, ASCII text, or a hex string as defined by the suboption parameters.
Step 8	**exit** **Example:** `Switch(config-dhcpv6-vs)# exit`	Returns to DHCP pool configuration mode.
Step 9	**exit** **Example:** `Switch(config-dhcpv6)# exit`	Returns to global configuration mode.
Step 10	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 11	**ipv6 dhcp server** [*poolname* | **automatic**] [**rapid-commit**] [**preference** *value*] [**allow-hint**] **Example:** `Switch(config-if)# ipv6 dhcp server automatic`	Enables DHCPv6 server function on an interface. • *poolname*—(Optional) User-defined name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0). • **automatic**—(Optional) Enables the system to automatically determine which pool to use when allocating addresses for a client. • **rapid-commit**—(Optional) Allows two-message exchange method. • **preference** *value*—(Optional) Configures the preference value carried in the preference option in the advertise message sent by the server. The range is from 0 to 255. The preference value default is 0.

	Command or Action	Purpose
		• **allow-hint**—(Optional) Specifies whether the server should consider client suggestions in the SOLICIT message. By default, the server ignores client hints.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Step 13	Do one of the following:
	• **show ipv6 dhcp pool**
	• **show ipv6 dhcp interface**
	Example:
	`Switch# show ipv6 dhcp pool`
	or
	`Switch# show ipv6 dhcp interface`
		• Verifies DHCPv6 pool configuration.
		• Verifies that the DHCPv6 server function is enabled on an interface.

Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Enabling DHCPv6 Server Function: Example, on page 53

Enabling DHCPv6 Client Function

To enable the DHCPv6 client on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode, and specifies the interface to configure.
Step 4	**ipv6 address dhcp** [**rapid-commit**] **Example:** `Switch(config-if)# ipv6 address dhcp rapid-commit`	Enables the interface to acquire an IPv6 address from the DHCPv6 server. **rapid-commit**—(Optional) Allow two-message exchange method for address assignment.
Step 5	**ipv6 dhcp client request** [**vendor-specific**] **Example:** `Switch(config-if)# ipv6 dhcp client request vendor-specific`	(Optional) Enables the interface to request the vendor-specific option.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show ipv6 dhcp interface** **Example:** `Switch# show ipv6 dhcp interface`	Verifies that the DHCPv6 client is enabled on an interface.

Related Topics

Enabling DHCPv6 Client Function: Example, on page 53

Configuration Examples for IPv6 Unicast Routing

Configuring IPv6 Addressing and Enabling IPv6 Routing: Example

This example shows how to enable IPv6 with both a link-local address and a global address based on the IPv6 prefix 2001:0DB8:c18:1::/64. The EUI-64 interface ID is used in the low-order 64 bits of both addresses. Output from the **show ipv6 interface** EXEC command is included to show how the interface ID (20B:46FF:FE2F:D940) is appended to the link-local prefix FE80::/64 of the interface.

```
Switch(config)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet0/11
```

```
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
Switch# show ipv6 interface gigabitethernet0/11
GigabitEthernet0/11 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
  2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/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
  Hosts use stateless autoconfig for addresses.
```

Related Topics

Configuring IPv6 Addressing and Enabling IPv6 Routing , on page 8

Configuring Default Router Preference: Example

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

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ipv6 nd router-preference high
Switch(config-if)# end
```

Related Topics

Configuring Default Router Preference, on page 20

Enabling an HSRP Group for IPv6: Example

This example shows how to activate HSRP for IPv6 for group 1 on a port. The IP address used by the hot standby group is learned by using HSRP for IPv6.

Note This procedure is the minimum number of steps required to enable HSRP for IPv6. Other configurations are optional.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ipv6 autoconfig
Switch(config-if)# end
Switch# show standby
```

Related Topics

Enabling an HSRP Group for IPv6, on page 33

Enabling DHCPv6 Server Function: Example

This example shows how to configure a pool called *engineering* with an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool engineering
Switch(config-dhcpv6)#address prefix 2001:1000::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *testgroup* with three link-addresses and an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool testgroup
Switch(config-dhcpv6)# link-address 2001:1001::0/64
Switch(config-dhcpv6)# link-address 2001:1002::0/64
Switch(config-dhcpv6)# link-address 2001:2000::0/48
Switch(config-dhcpv6)# address prefix 2001:1003::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *350* with vendor-specific options:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool 350
Switch(config-dhcpv6)# address prefix 2001:1005::0/48
Switch(config-dhcpv6)# vendor-specific 9
Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::1
Switch(config-dhcpv6-vs)# suboption 2 ascii "IP-Phone"
Switch(config-dhcpv6-vs)# end
```

Related Topics

Enabling DHCPv6 Server Function (CLI), on page 48

Enabling DHCPv6 Client Function: Example

This example shows how to acquire an IPv6 address and to enable the rapid-commit option:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# ipv6 address dhcp rapid-commit
```

Related Topics

Enabling DHCPv6 Client Function, on page 50

Configuring IPv6 ICMP Rate Limiting: Example

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
```

Related Topics

Configuring IPv6 ICMP Rate Limiting, on page 22

Configuring Static Routing for IPv6: Example

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 gigabitethernet 1/0/1 130
```

Related Topics

Configuring Static Routing for IPv6, on page 23

Configuring RIP for IPv6: Example

This example shows how to enable the RIP routing process *cisco* with a maximum of eight equal-cost routes and to enable it on an interface:

```
Switch(config)# ipv6 router rip cisco
Switch(config-router)# maximum-paths 8
Switch(config)# exit
Switch(config)# interface gigabitethernet2/0/11
Switch(config-if)# ipv6 rip cisco enable
```

Related Topics

Configuring RIP for IPv6, on page 25

Displaying IPv6: Example

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>
```

Related Topics

Displaying IPv6, on page 46