CHAPTER 1

Configuring IPv6 Unicast Routing

This chapter describes how to configure IPv6 unicast routing on the Catalyst 3750-E or 3560-E Catalyst 3750-X or 3560-X switch.

For information about configuring IPv4 unicast routing, see Chapter 1, "Configuring IP Unicast Routing."For information about configuring IPv6 Multicast Listener Discovery (MLD) snooping, see Chapter 1, "Configuring IPv6 MLD Snooping." For information on configuring IPv6 access control lists (ACLs) see Chapter 1, "Configuring IPv6 ACLs."

Note To use all IPv6 features in this chapter, the switch or stack master must be running the IP services feature set. Switches running the IP base feature set support only IPv6 static routing and RIP for IPv6. Switches running the LAN base feature set support only IPv6 host functionality.

Unless otherwise noted, the term *switch* refers to a 3750-X or 3560-X standalone switch and to a Catalyst 3750-X switch stack.

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

This chapter consists of these sections:

Understanding IPv6

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.

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

IPv6 Addresses

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

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

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

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

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

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

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

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

In the "Information About Implementing Basic Connectivity for IPv6" chapter, these sections apply to the switch:

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

Supported IPv6 Unicast Routing Features

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

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

The switch provides IPv6 routing capability over native Ethernet Inter-Switch Link (ISL) or 802.1Q trunk ports for static routes, Routing Information Protocol (RIP) for IPv6, and Open Shortest Path First (OSPF) Version 3 Protocol. It supports up to 16 equal-cost routes and can simultaneously forward IPv4 and IPv6 frames at line rate.

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. The switch does not support path MTU discovery for multicast packets.

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.

First Hop Security in IPv6

This section provides information about configuring the functions that comprise the first hop security (FHS) feature in IPv6.

Note First Hop Security in IPv6 is not supported on EtherChannels.

For more information on FHS, see this URL:

http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6_fhsec/configuration/xe-3s/ip6f-xe-3s-book.html

The functions available under FHS are also called as IPv6 policies. Policies can be applied at the interface or VLAN level. IPv6 policies provide policy database services to features with regard to storing and accessing those policies. Every time a policy is configured, the attributes of the policy are stored in the software policy database. The policy is then applied to an interface and the software policy database entry is updated to include this interface to which the policy is applied. You can use the following IPv6 policies:

Prerequisites for implementing first hop security

- You have configured the necessary IPv6 enabled SDM template.
- You should be familiar with the IPv6 neighbor discovery feature. For information, see "Implementing IPv6 Addressing and Basic Connectivity" chapter of the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Restrictions for implementing first hop security

Although visible in the command-line help strings, the IPv6 first hop security (FHS) is not supported on the Catalyst 3750-G and 3750v2 switches. The command-line help strings are visible on these switches to support the FHS feature in a mixed switch stack scenario where one of these switches could become a master.

IPv6 Snooping

IPv6 snooping acts as a container policy that enables most of the features available with FHS in IPv6. For more information, see the "Configuring an IPv6 Snooping Policy" section on page 1-21.

IPv6 First-Hop Security Binding Table

A database table of IPv6 neighbors connected to the switch is created from multiple sources of information, For example, Neighbor Discovery Protocol (NDP) snooping and Dynamic Host Configuration Protocol (DHCP) snooping. This database or binding table is used by various IPv6 guard features such as IPv6 Neighbor Discovery (ND) inspection (to validate the link-layer address (LLA)), per-port address limit (to validate the IPv4 or IPv6 addresses), and IPv6 device tracking (to prefix binding of the neighbors to prevent spoofing and redirect attacks).

These categories of traffic carry information that the binding table snoops for:

- ND traffic—For more information, see the "NDP Address Gleaning" section on page 1-6.
- DHCP traffic—For more information, see the "IPv6 DHCP Address Gleaning" section on page 1-6.
- Data traffic—For more information, see the "IPv6 DHCP Address Gleaning" section on page 1-6.

NDP Address Gleaning

The NDP address gleaning feature is enabled by default when you configure the **ipv6 snooping policy** global configuration command. To disable this function, enter the **no protocol ndp** global configuration command and attach the policy to the target port or VLAN.

IPv6 DHCP Address Gleaning

The IPv6 DHCP address gleaning feature provides the ability to extract addresses from DHCP messages and populate the binding table. The switch extracts address binding information from the following types of DHCPv6 exchanges (using User Datagram Protocol (UDP), ports 546 and 547):

- DHCP-REQUEST
- DHCP-CONFIRM
- DHCP- RENEW
- DHCP-REBIND
- DHCP-REPLY
- DHCP-RELEASE
- DHCP-DECLINE

After a switch receives a DHCP-REQUEST message from a client, one of the following can happen:

- The switch receives a DHCP-REPLY message from the DHCP server and a binding table entry is created in the REACHABLE state and completed. The reply contains the IP address and the MAC address in the Layer 2 DMAC field.

Creating an entry in the binding table allows the switch to learn addresses assigned by DHCP. A binding table can have one of the following states:

  - INCOMPLETE—Address resolution is in progress and the link-layer address is not yet known.
  - REACHABLE—The table is known to be reachable within the last reachable time interval.
  - STALE—The table requires re-resolution.
  - SEARCH—The feature creating the entry does not have the Layer 2 address and requests the binding table to search for the Layer 2 address.
  - VERIFY—The Layer 2 and Layer 3 addresses are known and a duplicate address detection (DAD) Neighbor solicitation (NS) unicast message is sent to the Layer 2 and Layer 3 destinations to verify the addresses.
  - DOWN—The interface from which the entry was learned is down, preventing verification.

- The DHCP server sends a DHCP-DECLINE or DHCP release message and the entry is deleted.
- The client sends a DHCP-RENEW message to the server that allocated the address or a DHCP-REBIND message to any server and the lifespan of the entry is extended.
- The server does not reply and the session is timed-out.

To enable this feature, configure a policy using the **ipv6 snooping policy** *policy-name* global configuration command. For more information, see the "Configuring an IPv6 Snooping Policy" section on page 1-21.

You can configure a policy and attach it to a DHCP guard to prevent the binding table from being filled with forged DHCP messages. For more information, see the "IPv6 DHCP Guard" section on page 1-9 and "Configuring IPv6 DHCP Guard" section on page 1-23.

IPv6 Binding Table Recovery Mechanism

The IPv6 first-hop security binding table recovery feature recovers the missing binding table entries when the resolution for a destination address fails in the destination guard. Upon a failure, a binding table entry is recovered by querying the DHCP server or the destination host depending on the configuration.

The recovery mechanism blocks any data traffic sourced from an unknown source, that is, a source not already specified in the binding table and previously learned by using NDP or Dynamic Host Configuration Protocol (DHCP) gleaning.

For detailed information about IPv6 binding table recovery, see the "IPv6 First-Hop Security Binding Table" chapter of the *Cisco IOS IPv6 Configuration Guide Library* on Cisco.com.

IPv6 Data Address Gleaning

The IPv6 data address gleaning feature provides the ability to extract addresses from redirected data traffic, to discover neighbors, and to populate binding tables.

When a port receives a data packet where the binding is unknown, that is, the neighbor is in an INCOMPLETE state and the link-layer address is not yet known, the switch sends a DAD NS NDP unicast message to the port from which the data packet was received.

After the host replies with a DAD Neighbor Advertisement (NA) NDP message, the binding table is updated and a private VLAN ACL (PVACL) is installed in the hardware for this binding.

If the host does not reply with a DAD NA, after the binding table timer expires, the hardware is notified and any resources associated with that binding are released.

To enable this feature, configure a policy with **data-glean** and attach the policy to a target port. To debug the policy, use the **debug ipv6 snooping** privileged EXEC command.

IPv6 ND 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 SA 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 Device Tracking

The IPv6 device tracking feature provides IPv6 host liveness tracking so that a neighbor table can be updated when an IPv6 host disappears. The feature tracks the liveness of the neighbors connected through the Layer 2 switch on regular basis in order to revoke network access privileges as they become inactive.

IPv6 Port-Based Access List Support

The IPv6 port-based access list (PACL) feature provides the ability to provide access control (permit or deny) on Layer 2 switch ports for IPv6 traffic. IPv6 PACLs are similar to IPv4 PACLs, which provide access control on Layer 2 switch ports for IPv4 traffic.

With Catalyst 3750-E, 3750X, 3560E, 3560-X, 3750v2, and 3560 v2 switches, this feature is supported in hardware and only in ingress direction. In a mixed stack scenario where the stack has a switch that does not support IPv6 FHS, the VLAN target is disabled on the whole switch for security. Port targets are allowed on the IPv6 FHS-capable ports of the switch. If a nonsupporting switch becomes the stack master, the IPv6 FHS functions are still supported on the IPv6 FHS-capable ports of the switch.

Access lists determine which traffic is blocked and which traffic is forwarded at switch interfaces and allow filtering based on source and destination addresses, inbound and outbound, to a specific interface. Each access list has an implicit deny statement at the end. To configure an IPv6 PACL, you have to create an IPv6 access list and then configure the PACL mode on the specified IPv6 Layer 2 interface.

PACL can filter ingress traffic on Layer 2 interfaces based on Layer 3 and Layer 4 header information or non-IP Layer 2 information.

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.

For detailed information about RA guard, see the " IPv6 RA Guard" chapter of the *Cisco IOS IPv6 Configuration Guide Library* on Cisco.com.

IPv6 DHCP Guard

You can use the DHCP guard to prevent forged messages from being entered in the binding table. The DHCP guard blocks DHCP server messages when they are received on ports that are not explicitly configured as facing a DHCP server or DHCP relay.

To use this feature, configure a policy and attach it to a DHCP guard. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.

For detailed information about IPv6 DHCP Guard, see the "IPv6 DHCP Guard" chapter of the *Cisco IOS IPv6 Configuration Guide Library* on Cisco.com.

IPv6 Source Guard

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

The IPv6 source guard feature provides the ability to use the IPv6 binding table to install PACLs to prevent a host from sending packets with an invalid IPv6 source address.

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

Note The IPv6 PACL feature is supported only in the ingress direction; it is not supported in the egress direction.

The following restrictions apply:

- 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.
- IPv6 source guard is not supported on EtherChannels.

For information about configuring IPv6 access lists, see the "Implementing Traffic Filters and Firewalls for IPv6 Security" chapter of the *Cisco IOS IPv6 Configuration 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 nontopologically 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 detailed information about prefix guard, see the "IPv6 Prefix Guard" chapter of the *Cisco IOS IPv6 Configuration Guide Library* on Cisco.com.

Pv6 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 detailed information about 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.

For detailed information about Neighbor Discovery multicast suppress, see the "IPv6 Neighbor Discovery Multicast Suppress" chapter of the *Cisco IOS IPv6 Configuration Guide Library* on Cisco.com and "Configuring IPv6 DHCP Guard" section on page 1-23.

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.

Default Router Preference

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

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

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

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

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

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

IPv6 Applications

The switch has IPv6 support for these applications:

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

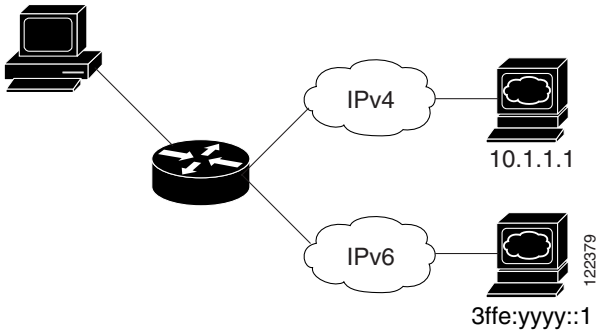
For more information about managing these applications, see the "Managing Cisco IOS Applications over IPv6" chapter and the "Implementing IPv6 Addressing and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Dual IPv4 and IPv6 Protocol Stacks

You must use the dual IPv4 and IPv6 template to allocate hardware memory usage to both IPv4 and IPv6 protocols.

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

Figure 1-1 Dual IPv4 and IPv6 Support on an Interface



Use the dual IPv4 and IPv6 switch database management (SDM) template to enable IPv6 routing dual stack environments (supporting both IPv4 and IPv6). For more information about the dual IPv4 and IPv6 SDM template, see Chapter 1, "Configuring SDM Templates."

The dual IPv4 and IPv6 templates allow the switch to be used in dual stack environments.

- If you try to configure IPv6 without first selecting a dual IPv4 and IPv6 template, a warning message appears.
- In IPv4-only environments, the switch routes IPv4 packets and applies IPv4 QoS and ACLs in hardware. IPv6 packets are not supported.
- In dual IPv4 and IPv6 environments, the switch routes both IPv4 and IPv6 packets and applies IPv4 QoS in hardware.
- The switch supports QoS for both IPv4and IPv6 traffic.
- If you do not plan to use IPv6, do not use the dual stack template because this template results in less hardware memory capacity for each resource.

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

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.

Beginning with Cisco IOS Release 12.2(58)SE, the switch supports these features:

- DHCPv6 bulk-lease query

  DHCPv6 bulk-lease query allows a client to request information about DHCPv6 bindings. This functionality adds new query types and allows the bulk transfer of DHCPv6 binding data through TCP. Bulk transfer of DHCPv6 binding data is useful when the relay server switch is rebooted and the relay server has lost all the binding information. After the reboot, the relay server automatically generates a bulk-lease query to get the binding information from the DHCP server.

- DHCPv6 relay source configuration

  The DHCPv6 server replies to the source address of the DHCP relay agent. Typically, messages from a DHCPv6 relay agent show the source address of the interface from which they are sent. You can use the DHCPv6 relay source configuration feature to configure a more stable address (such as a loopback interface) as the source address for messages from the relay agent. You can configure the source address globally for the switch or for a specific interface. An address configured on an interface takes precedence over one configured globally.

For more information and to configure these features, see the *Cisco IOS IPv6 Configuration Guide, Release 12.4*.

This document 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.

Static Routes for IPv6

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

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

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 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 IP-services feature set supports Open Shortest Path First (OSPF) for IPv6, a link-state protocol for IP. For more information, see the "Implementing OSFP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

OSPFv3 Graceful Restart

Beginning with Cisco IOS Release 12.2(58)SE, switches running the IP Services feature set support the graceful restart feature in OSPFv3. This 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 OSFP 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.

For more information, see the "Implementing OSPFv3" chapter of the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

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.

For more information, see the OSPFv3 Authentication Support with IPsec section of the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

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 support IPv6 EIGRP stub routing only.

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.

HSRP for IPv6

Switches running the IP Services feature set support the Hot Standby Router Protocol (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.

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.

SNMP and Syslog Over IPv6

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

SNMP and syslog over IPv6 provide these features:

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

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

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

- Verifies SNMP Manager feature works with IPv6 transport

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

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

HTTP(S) Over IPv6

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

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

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

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

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

Unsupported IPv6 Unicast Routing Features

The switch does not support these IPv6 features:

- IPv6 policy-based routing
- IPv6 virtual private network (VPN) routing and forwarding (VRF) table support
- 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 unicast reverse-path forwarding
- IPv6 general prefixes

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:

ICMPv6 redirect functionality is not supported for IPv6 host routes (routes used to reach a specific host) or for IPv6 routes with masks greater than 64 bits. The switch cannot redirect hosts to a better first-hop router for a specific destination that is reachable through a host route or through a route with masks greater than 64 bits.

- Load balancing using equal cost and unequal cost routes is not supported for IPv6 host routes or for IPv6 routes with a mask greater than 64 bits.

- The switch cannot forward SNAP-encapsulated IPv6 packets.

Note There is a similar limitation for IPv4 SNAP-encapsulated packets, but the packets are dropped at the switch and are not forwarded.

- The switch routes IPv6-to-IPv4 and IPv4-to-IPv6 packets in hardware, but the switch cannot be an IPv6-to-IPv4 or IPv4-to-IPv6 tunnel endpoint.
- Bridged IPv6 packets with hop-by-hop extension headers are forwarded in software. In IPv4, these packets are routed in software, but bridged in hardware.
- Interface counters for IPv6 traffic include software-forwarded traffic only; hardware-switched traffic is excluded.
- In addition to the normal SPAN and RSPAN limitations defined in the software configuration guide, these limitations are specific to IPv6 packets:
  - When you send RSPAN IPv6-routed packets, the source MAC address in the SPAN output packet can be incorrect.
  - When you send RSPAN IPv6-routed packets, the destination MAC address can be incorrect. Normal traffic is not affected.
- The switch cannot apply QoS classification or policy-based routing on source-routed IPv6 packets in hardware.
- The switch cannot generate ICMPv6 `Packet Too Big` messages for multicast packets.

IPv6 and Switch Stacks

The switch supports IPv6 forwarding across the stack and IPv6 host functionality on the stack master. The stack master runs the IPv6 unicast routing protocols and computes the routing tables. Using distributed CEF (dCEF), the stack master downloads the routing table to the stack member switches. They receive the tables and create hardware IPv6 routes for forwarding. The stack master also runs all IPv6 applications.

Note To route IPv6 packets in a stack, all switches in the stack should be running the IP Services feature set.

If a new switch becomes the stack master, it recomputes the IPv6 routing tables and distributes them to the member switches. While the new stack master is being elected and is resetting, the switch stack does not forward IPv6 packets. The stack MAC address changes, which also changes the IPv6 address. When you specify the stack IPv6 address with an extended unique identifier (EUI) by using the **ipv6 address** *ipv6-prefix/prefix length* **eui-64** interface configuration command, the address is based on the interface MAC address. See the "Configuring IPv6 Addressing and Enabling IPv6 Routing" section on page 1-18.

If you configure the persistent MAC address feature on the stack and the stack master changes, the stack MAC address does not change for approximately 4 minutes. For more information, see the "Enabling Persistent MAC Address" section on page 1-24 in Chapter 1, "Managing Switch Stacks."

The IPv6 stack master and members have these features:

- Stack master:
  - Runs IPv6 routing protocols
  - Generates routing tables

  - Distributes CEFv6 routing tables to stack members that use dCEFv6
  - Runs IPv6 host functionality and IPv6 applications
- Stack member (must be running the IP services feature set):
  - Receives CEFv6 routing tables from the stack master
  - Programs the routes into hardware

Note IPv6 packets are routed in hardware across the stack if the packet does not have exceptions (IPv6 Options) and the switches in the stack have not run out of hardware resources.

  - Flushes the CEFv6 tables on master re-election

Configuring IPv6

Default IPv6 Configuration

Table 1-1 Default IPv6 Configuration

Feature	Default Setting
SDM template	Default desktop.
IPv6 routing	Disabled globally and on all interfaces

Table 1-1 Default IPv6 Configuration (continued)

Feature	Default Setting
CEFv6 or dCEFv6	Disabled (IPv4 CEF and dCEF are enabled by default) **Note** When IPv6 routing is enabled, CEFv6 and dCEF6 are automatically enabled.
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.
- Not all features discussed in this chapter are supported by the switch. See the "Unsupported IPv6 Unicast Routing Features" section on page 1-15.
- 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:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**sdm prefer dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**}	Selects an SDM template that supports IPv4 and IPv6. • **default**—Sets the switch to the default template to balance system resources. • **routing**—Sets the switch to the routing template to support IPv4 and IPv6 routing, including IPv4 policy-based routing. • **vlan**—Maximizes VLAN configuration on the switch with no routing supported in hardware.
Step 3	**end**	Returns to privileged EXEC mode.

	Action or Command	Purpose
Step 4	**reload**	Reloads the operating system.
Step 5	**configure terminal**	Enters global configuration mode after the switch reloads.
Step 6	**interface** *interface-id*	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 7	**no switchport**	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
Step 8	**ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **ipv6 address** *ipv6-address/prefix length* or **ipv6 address** *ipv6-address* **link-local** or **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 9	**exit**	Returns to global configuration mode.
Step 10	**ip routing**	Enables IP routing on the switch.
Step 11	**ipv6 unicast-routing**	Enables forwarding of IPv6 unicast data packets.
Step 12	**end**	Returns to privileged EXEC mode.
Step 13	**show ipv6 interface** *interface-id*	Verifies your entries.
Step 14	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

To remove an IPv6 address from an interface, use the **no ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **no ipv6 address** *ipv6-address* **link-local** interface configuration command. To remove all manually configured IPv6 addresses from an interface, use the **no ipv6 address** interface configuration command without arguments. To disable IPv6 processing on an interface that has not been explicitly configured with an IPv6 address, use the **no ipv6 enable** interface configuration command. To globally disable IPv6 routing, use the **no ipv6 unicast-routing** global configuration command.

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)# sdm prefer dual-ipv4-and-ipv6 default
Switch(config)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
Switch# show ipv6 interface gigabitethernet1/0/11
```

```
GigabitEthernet1/0/11 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.
```

Configuring First Hop Security in IPv6

Configuring an IPv6 Snooping Policy

	Action or Command	Purpose
Step 1	**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** | **dhcp** | **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 of addresses 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. – **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.

To attach a snooping policy to an interface or VLAN, complete the following steps:

	Action or Command	Purpose
Step 1	**enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	**interface** *type number*	Specifies an interface type and number, and enters the interface configuration mode.
Step 4	**switchport**	Configures the interface as a Layer 2 port.
Step 5	**exit**	Exits the interface configuration mode.
Step 6	**ipv6 snooping attach-policy** *policy-name* or **vlan configuration** *vlan list* **ipv6 snooping attach-policy** *policy-name*	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 7	**show ipv6 snooping policy** *policy-name*	Displays the snooping policy configuration.
Step 8	**show ipv6 neighbors binding**	Displays the binding table entries populated by the snooping policy.

Configuring IPv6 Neighbor Discovery Multicast Suppress Policy

To configure Neighbor Discovery Multicast Suppress policy on a device, complete the following steps:

	Action or Command	Purpose
Step 1	**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.
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.

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

	Action or Command	Purpose
Step 1	**enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal**	Enters the global configuration mode.
Step 3	**interface** *type number*	Specifies an interface type and number, and places the device in interface configuration mode.

	Action or Command	Purpose
Step 4	**ipv6 nd suppress attach-policy** [*policy-name* [**vlan** {**add** | **except** | **none** | **remove** | **all**} *vlan* [*vlan1, vlan2, vlan3...*]]] or **vlan configuration** *vlan-id*	Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.
Step 5	**exit**	Exits interface configuration mode.

Configuring IPv6 DHCP Guard

	Action or Command	Purpose
Step 1	**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*	Creates a policy in global configuration mode and enters the DHCP guard policy global configuration mode.
Step 4	[**default** | **device-role** [**client** | **server**] | **no** | **exit** | **trusted-port**]	Configures the parameters for the DHCP guard policy. • (Optional) **default**—Set a command to its defaults. • (Optional) **device-role** [**client** | **server**]—Qualifies the role of the device attached to the port. – **client**—Specifies that the attached device is a client. This is the default. Any server messages are dropped on this port. – **server**—Specifies that the attached device is a DHCP server. Server messages are allowed on this port. • (Optional) **no**—Removes the configured policy parameters. • (Optional) **exit**—Exits the DHCP guard policy global configuration mode. • (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	**interface** *type number*	Specifies an interface type and number and enters the interface configuration mode.
Step 7	**ipv6 dhcp guard attach-policy** *policy-name* Or **vlan configuration** *vlan-id*	Attaches the DHCP guard policy to an interface or VLAN.
Step 8	**show ipv6 dhcp guard policy** *policy-name*	Displays the DHCP guard policy configuration.

Configuring IPv6 Source Guard

	Action or Command	Purpose
Step 1	**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 source guard policy name and enters the source guard policy configuration mode.
Step 4	**permit link-local**	Allows all data traffic that is sourced by a link-local address.
Step 5	**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.
Step 6	**ipv6 source-guard** [**attach-policy** *policy-name*]	Specifies the policy name. (Optional) **attach-policy** *policy-name*—Filters based on the policy name
Step 7	**exit**	Exits the source guard policy configuration mode.
Step 8	**show ipv6 source-guard policy** *policy name*	Shows the policy configuration and all the interfaces where the policy is applied.

Configuration Examples for Implementing First Hop Security in IPv6

This example shows you how to attach a snooping policy to a VLAN and to configure an RA trusted router port and DHCP trusted server port:

```
Switch(config)# vlan configuration 100
Switch(config-vlan-config)# ipv6 snooping
Switch(config-vlan-config)# exit

Switch(config)# ipv6 nd raguard policy router
Switch(config-nd-raguard)# device-role router
Switch(config-nd-raguard)# exit

Switch(config)# ipv6 dhcp guard policy server
Switch(config-dhcp-guard)# device-role server
Switch(config-dhcp-guard)# exit
```

2/1/2 is a router-facing port:

```
Switch(config)# interface fastethernet 2/1/2
Switch(config-if)# switchport
Switch(config-if)# swithport access vlan 100
Switch(config-if)# ipv6 nd raguard attach-policy router
Switch(config-if)# exit
```

1/0/17 is a DHCP server-facing port:

```
Switch(config)# interface gigabitethernet 1/0/17
Switch(config-if)# switchport access vlan 100
Switch(config-if)# ipv6 dhcp guard attach-policy server
Switch(config-if)# exit
Switch(config)# exit

Switch# show ipv6 snooping policies
Target              Type  Policy              Feature        Target range
Gi1/0/17            PORT  server              DHCP Guard     vlan all
```

```
Te2/1/2              PORT  router              RA guard       vlan all
vlan 100             VLAN  default             Snooping       vlan all
```

This example shows you how to create a snooping policy called *Test* and enable data address gleaning on it:

```
Switch(config)# ipv6 snooping policy Test
Switch(config-ipv6-snooping)# data-glean
Switch(config-ipv6-snooping)# device-role node
Switch(config-ipv6-snooping)# limit address-count 1
Switch(config-ipv6-snooping)# protocol dhcp
Switch(config-ipv6-snooping)# security-level glean
Switch(config-ipv6-snooping)# tracking enable
Switch(config-ipv6-snooping)# no trusted-port
Switch(config-ipv6-snooping)# exit
```

This example shows you how to configure snooping policy named *Test*, enable data address gleaning on the policy, and enable source guard where link-local addresses are permitted and global autoconfiguration addresses are denied entry:

```
Switch(config)# ipv6 snooping policy Test
Switch(config-ipv6-snooping)# data-glean
Switch(config-ipv6-snooping)# exit
Switch(config)# ipv6 source-guard policy Test
Switch(config-sisf-sourceguard)# permit link-local
Switch(config-sisf-sourceguard)# deny global-autoconf
Switch(config-sisf-sourceguard)# exit
```

This example shows you how to attach a snooping policy with source guard to an interface:

```
Switch(config)# interface gigabitethernet2/0/3
Switch(config-if)# ipv6 snooping attach-policy Test
Switch(config-if)# ipv6 source-guard attach-policy Test

Switch# show ipv6 source-guard policy Test
Policy Test configuration:
   permit link-local
   deny global-autoconf
Policy Test is applied on the following targets:
   Target  Type  Policy Feature      Target range
   Gi2/0/3 PORT  Test   Source guard vlan all
```

This example shows you how to configure a DHCP guard policy named *Test* and attach it to an interface:

```
Switch(config)# ipv6 dhcp-guard policy Test
Switch(config-dhcp-guard)# no trusted-port
Switch(config-dhcp-guard)# exit

Switch(config)# interface gigabitethernet2/0/3
Switch(config-if)# ipv6 dhcp guard attach-policy Test
Switch(config-if)# exit
OR
Switch(config)# vlan configuration 1-10
Switch(config-vlan-config)# ipv6 dhcp guard attach-policy Test
Switch(config-vlan-config)# exit

Switch# show ipv6 dhcp-guard policy Test
Dhcp guard policy: Test
Device Role: dhcp server
Target: Gi2/0/3 vlan 1 vlan 2 vlan 3 vlan 4 vlan 5 vlan 6 vlan 7 vlan 8 vlan 9 vlan 10
Max Preference: 255
Min Preference: 0
```

This example shows how you can enable the FHS feature on an interface or VLAN, without creating a snooping policy.

Note Creating a policy gives you the flexibility to configure as per your needs. If you enable the feature without creating a policy, then the default policy configuration is applied:

```
Switch(config)# interface GigabitEthernet1/0/9
Switch(config-if)# ipv6 nd inspection
Switch(config-if)# ipv6 nd raguard
Switch(config-if)# ipv6 snooping
Switch(config-if)# ipv6 dhcp guard
Switch(config-if)# ipv6 source-guard
Switch(config-if)# end
```

OR

```
Switch(config)# vlan configuration 1
Switch(config-vlan-config)# ipv6 nd inspection
Switch(config-vlan-config)# ipv6 nd raguard
Switch(config-vlan-config)# ipv6 dhcp guard
Switch(config-vlan-config)# ipv6 snooping
```

Note You cannot apply a source-guard policy to the VLAN.

For more examples, see the Configuration Examples for Implementing First Hop Security in IPv6 section of the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

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 the policy might dictate that hosts should prefer one of the routers.

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

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

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

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

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

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

Configuring IPv4 and IPv6 Protocol Stacks

Before configuring IPv6 routing, you must select an SDM template that supports IPv4 and IPv6. If not already configured, use the **sdm prefer dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**} [**desktop**] global configuration command to configure a template that supports IPv6. When you select a new template, you must reload the switch by using the **reload** privileged EXEC command so that the template takes effect.

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface to support both IPv4 and IPv6 and to enable IPv6 routing.

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ip routing**	Enables routing on the switch.
Step 3	**ipv6 unicast-routing**	Enables forwarding of IPv6 data packets on the switch.
Step 4	**interface** *interface-id*	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 5	**no switchport**	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
Step 6	**ip address** *ip-address mask* [**secondary**]	Specifies a primary or secondary IPv4 address for the interface.
Step 7	**ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **ipv6 address** *ipv6-address* **link-local** or **ipv6 enable**	Specifies a global IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. Specifies a link-local address on the interface to be used instead of the automatically configured link-local address when IPv6 is enabled 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 8	**end**	Returns to privileged EXEC mode.
Step 9	**show interface** *interface-id* **show ip interface** *interface-id* **show ipv6 interface** *interface-id*	Verifies your entries.
Step 10	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

To disable IPv4 routing, use the **no ip routing** global configuration command. To disable IPv6 routing, use the **no ipv6 unicast-routing** global configuration command. To remove an IPv4 address from an interface, use the **no ip address** *ip-address mask* interface configuration command. To remove an IPv6 address from an interface, use the **no ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **no ipv6 address** *ipv6-address* **link-local** interface configuration command. To remove all manually configured IPv6 addresses from an interface, use the **no ipv6 address** interface configuration command without arguments. To disable IPv6 processing on an interface that has not been explicitly configured with an IPv6 address, use the **no ipv6 enable** interface configuration command.

This example shows how to enable IPv4 and IPv6 routing on an interface:

```
Switch(config)# sdm prefer dual-ipv4-and-ipv6 default
Switch(config)# ip routing
Switch(config)# ipv6 unicast-routing
Switch(config)# interface fastethernet1/0/11
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.168.99.1 244.244.244.0
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
```

Configuring DHCP for IPv6 Address Assignment

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 — Aport-channel logical interface created by using the **interface port-channel port-channel-number** command.
- Before configuring DHCPv6, you must select a Switch Database Management (SDM) template that supports IPv4 and IPv6.
- 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.
- The DHCPv6 client, server, or relay agent runs only on the master switch. When there is a stack master reelection, the new master switch retains the DHCPv6 configuration. However, the local RAM copy of the DHCP server database lease information is not retained.

Enabling DHCPv6 Server Function

Beginning in privileged EXEC mode, follow these steps to enable the DHCPv6 server function on an interface:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 dhcp pool** *poolname*	Enters DHCP pool configuration mode, and defines 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 3	**address prefix** *IPv6-prefix* **lifetime** {*t1 t1* | **infinite**}	(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 4	**link-address** *IPv6-prefix*	(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.

	Action or Command	Purpose
Step 5	**vendor-specific** *vendor-id*	(Optional) Enters vendor-specific configuration mode and enter a vendor-specific identification number. This number is the vendor IANA Private Enterprise Number. The range is 1 to 4294967295.
Step 6	**suboption** *number* {**address** *IPv6-address* | **ascii** *ASCII-string* | **hex** *hex-string*}	(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 7	**exit**	Returns to DHCP pool configuration mode.
Step 8	**exit**	Returns to global configuration mode.
Step 9	**interface** *interface-id*	Enters interface configuration mode, and specifies the interface to configure.
Step 10	**ipv6 dhcp server** [*poolname* | **automatic**] [**rapid-commit**] [**preference** *value*] [**allow-hint**]	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) 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. • **allow-hint**—(Optional) Specifies whether the server should consider client suggestions in the SOLICIT message. By default, the server ignores client hints.
Step 11	**end**	Returns to privileged EXEC mode.
Step 12	**show ipv6 dhcp pool** or **show ipv6 dhcp interface**	Verifies DHCPv6 pool configuration. Verifies that the DHCPv6 server function is enabled on an interface.
Step 13	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

To delete a DHCPv6 pool, use the **no ipv6 dhcp pool** *poolname* global configuration command. 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.

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
```

Enabling DHCPv6 Client Function

Beginning in privileged EXEC mode, follow these steps to enable DHCPv6 client function on an interface:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id*	Enters interface configuration mode, and specifies the interface to configure.
Step 3	**ipv6 address dhcp** [**rapid-commit**]	Enables the interface to acquire an IPv6 address from the DHCPv6 server. **rapid-commit**—(Optional) Allows two-message exchange method for address assignment.
Step 4	**ipv6 dhcp client request** [**vendor-specific**]	(Optional) Enables the interface to request the vendor-specific option.
Step 5	**end**	Returns to privileged EXEC mode.
Step 6	**show ipv6 dhcp interface**	Verifies that the DHCPv6 client is enabled on an interface.

To disable the DHCPv6 client function, use the **no ipv6 address dhcp** interface configuration command. To remove the DHCPv6 client request, use the **no ipv6 address dhcp client request** interface configuration command.

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
```

This document 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.

Configuring IPv6 ICMP Rate Limiting

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

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

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 icmp error-interval** *interval* [*bucketsize*]	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 3	**end**	Returns to privileged EXEC mode.
Step 4	**show ipv6 interface** [*interface-id*]	Verifies your entries.
Step 5	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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

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

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

Configuring CEF and dCEF for IPv6

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology to improve network performance. CEF implements an advanced IP lookup 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. In a Catalyst 3750-E switch stack, the hardware uses distributed CEF (dCEF) in the stack. IPv4 CEF and dCEF are enabled by default. IPv6 CEF and dCEF are disabled by default, but automatically enabled when you configure IPv6 routing.

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.

To disable IPv6 CEF or distributed CEF, use the **no ipv6 cef** or **no ipv6 cef distributed** global configuration command. To reenable IPv6 CEF or dCEF if it has been disabled, use the **ipv6 cef** or **ipv6 cef distributed** global configuration command. You can verify the IPv6 state by entering the **show ipv6 cef** privileged EXEC command.

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

Configuring Static Routing for IPv6

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

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

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*]	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 3	**end**	Returns to privileged EXEC mode.

	Action or Command	Purpose
Step 4	**show ipv6 static** [*ipv6-address* | *ipv6-prefix/prefix length*] [**interface** *interface-id*] [**recursive**] [**detail**] or **show ipv6 route static** [*updated*]	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 5	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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

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

```
Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130
```

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

Configuring RIP for IPv6

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.

Beginning in privileged EXEC mode, follow these required and optional steps to configure IPv6 RIP:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 router rip** *name*	Configures an IPv6 RIP routing process, and enters router configuration mode for the process.
Step 3	**maximum-paths** *number-paths*	(Optional) Defines the maximum number of equal-cost routes that IPv6 RIP can support. The range is from 1 to 64, and the default is four routes.
Step 4	**exit**	Returns to global configuration mode.
Step 5	**interface** *interface-id*	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 6	**ipv6 rip** *name* **enable**	Enables the specified IPv6 RIP routing process on the interface.

	Action or Command	Purpose
Step 7	**ipv6 rip** *name* **default-information** {**only** | **originate**}	(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. • **only**— Originates the default route, but suppress all other routes in the updates sent on this interface. • **originate**— Originates the default route in addition to all other routes in the updates sent on this interface.
Step 8	**end**	Returns to privileged EXEC mode.
Step 9	**show ipv6 rip** [*name*] [**interface** *interface-id*] [**database**] [**next-hops**] or **show ipv6 route rip** [*updated*]	Displays information about current IPv6 RIP processes. Displays the current contents of the IPv6 routing table.
Step 10	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

To disable a RIP routing process, use the **no ipv6 router rip** *name* global configuration command. To disable the RIP routing process for an interface, use the **no ipv6 rip** *name* interface configuration command.

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
```

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

Configuring OSPF for IPv6

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.

Beginning in privileged EXEC mode, follow these required and optional steps to configure IPv6 OSPF:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 router ospf** *process-id*	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 3	**area** *area-id* **range** {*ipv6-prefix/prefix length*} [**advertise** | **not-advertise**] [**cost** *cost*]	(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) Metric or cost for this summary route, which is used during OSPF SPF calculation to determine the shortest paths to the destination. The value can be 0 to 16777215.
Step 4	**maximum paths** *number-paths*	(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 64, and the default is 16 paths.
Step 5	**exit**	Returns to global configuration mode.
Step 6	**interface** *interface-id*	Enters interface configuration mode, and specify the Layer 3 interface to configure.
Step 7	**ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*]	Enables OSPF for IPv6 on the interface. • **instance** *instance-id*—(Optional) Instance identifier.
Step 8	**end**	Returns to privileged EXEC mode.
Step 9	**show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-id*] or **show ipv6 ospf** [*process-id*] [*area-id*]	Displays information about OSPF interfaces. Displays general information about OSPF routing processes.
Step 10	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

To disable an OSPF routing process, use the no **ipv6 router ospf** *process-id* global configuration command. To disable the OSPF routing process for an interface, use the **no ipv6 ospf** *process-id* **area** *area-id* interface configuration command.

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.

Tuning LSA and SPF Timers for OSPFv3 Fast Convergence

Beginning in the privileged EXEC mode, follow these steps to tune LSA and SPF timers:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 router ospf** *process-id*	Enables OSPFv3 router configuration mode.
Step 3	**timers lsa arrival** *milliseconds*	Sets the minimum interval at which the software accepts the same LSA from OSPFv3 neighbors.
Step 4	**timers pacing flood** *milliseconds*	Configures LSA flood packet pacing.
Step 5	**timers pacing lsa-group** *seconds*	Changes the interval at which OSPFv3 LSAs are collected into a group and refreshed, checksummed, or aged.
Step 6	**timers pacing retransmission** *milliseconds*	Configures LSA retransmission packet pacing in OSPFv3.

Configuring LSA and SPF Throttling for OSPFv3 Fast Convergence

Beginning in the privileged EXEC mode, follow these steps to configure LSA and SPF throttling:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 router ospf** *process-id*	Enables OSPFv3 router configuration mode.
Step 3	**timers throttle spf** *spf-start spf-hold spf-max-wait*	Turns on SPF throttling.
Step 4	**timers throttle lsa** *start-interval hold-interval max-interval*	Sets rate-limiting values for OSPFv3 LSA generation.
Step 5	**timers lsa arrival** *milliseconds*	Sets the minimum interval at which the software accepts the same LSA from OSPFv3 neighbors.
Step 6	**timers pacing flood** *milliseconds*	Configures LSA flood packet pacing.

For more information, see the "Enabling Event Logging for LSA and SPF Rate Limiting," "Verifying OSPFv3 Configuration and Operation," and "Example: Configuring LSA and SPF Throttling for OSPFv3 Fast Convergence" sections of the *Implementing OSPFv3* chapter of the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Configuring IPSec on OSPFv3

Note To enable authentication and encryption, configure the IP Security (IPsec) secure socket application program interface (API) on OSPFv3.

For information about configuring IPsec, see the following sections of the *Cisco IOS IPv6 Configuration Library* on Cisco.com:

- Defining Authentication on an Interface
- Defining Encryption on an Interface
- Defining Authentication in an OSPFv3 Area
- Defining Encryption in an OSPFv3 Area
- Defining Authentication and Encryption for a Virtual Link in an OSPFv3 Area
- Verifying OSPFv3 Configuration and Operation

Configuring EIGRP for IPv6

By default, EIGRP for IPv6 is disabled. You can configure EIGRP for IPv6 on an interface. After configuring the router and the interface for EIGRP, enter the **no shutdown** privileged EXEC command to start EIGRP.

Note If EIGRP for IPv6 is not in shutdown mode, EIGRP might start running before you enter the EIRGP router-mode commands to configure the router and the interface.

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 IPv4 interfaces and to select a subset of those as passive interfaces. Use the **passive-interface default** command to make all interfaces 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. By default, the group has a virtual IPv6 link-local address that is 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.

For configuration guidelines when configuring HSRP for IPv6 with HSRPv1 and HSRPv2, see the "Default HSRP Configuration" section on page 1-5 and the "Troubleshooting HSRP for Mixed Stacks of Catalyst 3750-X, 3750-E and 3750 Switches" section on page 1-12.

For more information about HSRP for IPv6 and HSRPv2, see the Chapter 1, "Configuring HSRP and VRRP."

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

Beginning in privileged EXEC mode, follow these steps to enable HSRP version 2 on a Layer 3 interface:

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id*	Enters interface configuration mode, and enter the Layer 3 interface on which you want to specify the standby version.
Step 3	**standby version** {**1** | **2**}	Changes the HSRP version with 2. The default is 1.
Step 4	**end**	Returns to privileged EXEC mode.
Step 5	**show standby**	Verifies the configuration.
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling an HSRP Group for IPv6

Beginning in privileged EXEC mode, follow these steps to create or enable HSRP for IPv6 on a Layer 3 interface:

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

	Action or Command	Purpose
Step 3	**standby** [*group-number*] **ipv6** {*link-local-address* | **autoconfig**}	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*}]	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 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*	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**	Returns to privileged EXEC mode.
Step 7	**show standby** [*interface-id* [*group-number*]]	Verifies the configuration.
Step 8	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Use the **no standby** [*group-number*] **ipv6** interface configuration command to disable HSRP for IPv6.

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
```

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.

Configuring Multi-VRF CE

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

To understand multi-VRF CE, see the "Understanding Multi-VRF CE" section on page 1-77.

Multi-VRF CE can be configured for a particular IP address family (IPv4 or IPv6 or both). This section provides information about multi-VRF CE for IPv6.

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 1-2 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

Beginning in privileged EXEC mode, follow these steps to configure one or more VRFs.

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ipv6 unicast-routing**	Enables IPv6 unicast routing.
Step 3	**vrf definition** *vrf-name*	Names the VRF, and enters VRF configuration mode.
Step 4	**address-family** ipv6	Specifies the IPv6 address family and enter address family configuration mode.
Step 5	**rd** *route-distinguisher*	(Optional) 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*	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 5.
Step 7	**import map** *route-map*	(Optional) Associates a route map with the VRF.
Step 8	**interface** *interface-id*	Specifies the Layer 3 interface to be associated with the VRF, and enters the interface configuration mode. The interface can be a routed port or SVI.
Step 9	**vrf forwarding** *vrf-name*	Associates the VRF with the Layer 3 interface.
Step 10	**end**	Returns to privileged EXEC mode.
Step 11	**show vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*]	Verifies the configuration. Displays information about the configured VRFs.
Step 12	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Use the **no vrf definition** *vrf-name* global configuration command to delete a VRF and to remove all interfaces from it. Use the **no vrf forwarding** *vrf-name* interface configuration command to remove an interface from the VRF.

Configuring VRF-Aware Services

These services are VRF-Aware:

- Neighbor discovery
- Ping
- Hot Standby Router Protocol (HSRP)
- Traceroute
- FTP and TFTP

User Interface for Neighbor Discovery

Beginning in privileged EXEC mode, follow these steps to configure VRF-aware services for Neighbor discovery.

Action or Command	Purpose
show ipv6 neighbors vrf *vrf-name*	Displays IPv6 neighbor cache entries for a specified VRF.

User Interface for PING

Beginning in privileged EXEC mode, follow these steps to configure VRF-aware services for ping.

Action or Command	Purpose
ping vrf *vrf-name* **ipv6** *ipv6-address*	Verifies the address reachability for a specified VRF.

User Interface for HSRP

Beginning in privileged EXEC mode, follow these steps to configure VRF-aware services for HSRP.

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id*	Enters interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no switchport**	Removes the interface from Layer 2 configuration mode if it is a physical interface.
Step 4	**vrf forwarding** *vrf-name*	Configures VRF on the interface.
Step 5	**ipv6 address** *ipv6-address*	Enters the IPv6 address for the interface.
Step 6	**standby 1 ipv6** *ipv6-address*	Enables HSRP and configure the virtual IP address.
Step 7	**end**	Returns to privileged EXEC mode.

User Interface for Traceroute

Beginning in privileged EXEC mode, follow these steps to configure VRF-aware services for traceroute.

Action or Command	Purpose
traceroute vrf *vrf-name ipv6-address*	Specifies the name of a VPN VRF in which to find the destination address.

User Interface for FTP and TFTP

To specify the source IP address for FTP connections, use the **ip ftp source-interface** show mode command. To use the address of the interface where the connection is made, use the **no** form of this command.

	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ip ftp source-interface** *interface-type interface-number*	Specifies the source IP address for FTP connections.
Step 3	**end**	Returns to privileged EXEC mode.

To specify the IP address of an interface as the source address for TFTP connections, use the **ip tftp source-interface** show mode command. To return to the default, use the **no** form of this command.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**ip tftp source-interface** *interface-type interface-number*	Specifies the source IP address for TFTP connections.
Step 3	**end**	Returns to privileged EXEC mode.

Configuring a VPN Routing Session

Routing within the VPN can be configured with static routing or with any supported routing protocol (OSPF, EIGRP, or BGP).

Static Route

	Action or Command	Purpose
Step 1	**ipv6 route** [**vrf** *vrf-name*] *ipv6-prefix/prefix-length* {*ipv6-address* | **interface-type** *interface-number* [*ipv6-address*]}	Configures static routes specific to VRF.

Routing Protocols

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** a*utonomous-system-number* address-family configuration mode command.

Beginning in privileged EXEC mode, follow these steps to configure OSPF in the VPN:

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

Use the **no router ospfv3** *process-id* global configuration command to disassociate the VPN forwarding table from the OSPF routing process.

Configuring BGP PE to CE Routing Sessions

Beginning in privileged EXEC mode, follow these steps to configure a BGP PE to CE routing session:

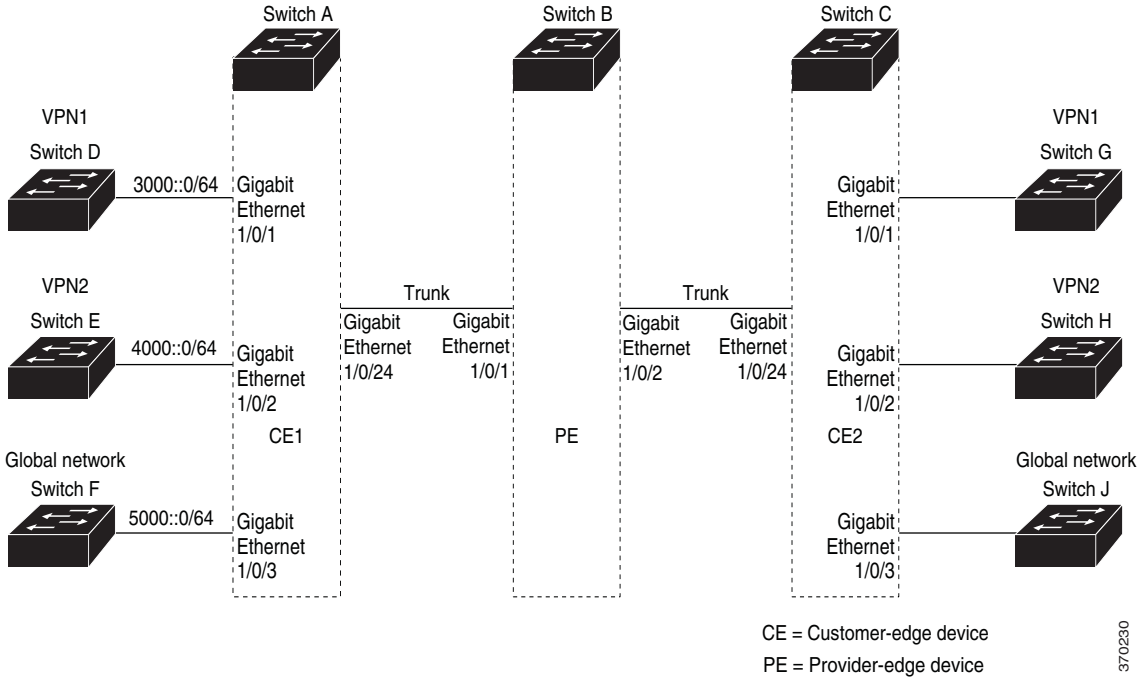
	Action or Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**router bgp** *autonomous-system-number*	Configures the BGP routing process with the AS number passed to other BGP routers, and enters router configuration mode.
Step 3	**bgp router-id** *router-id*	(Optional) Configures a fixed 32-bit router id as the identifier of the local router running BGP.
Step 4	**redistribute ospf** *process-id*	Sets the switch to redistribute OSPF internal routes.
Step 5	**address family ipv6 vrf** *vrf-name*	Defines BGP parameters for PE to CE routing sessions, and enters VRF address-family mode.
Step 6	**network** *ipv6 network-number*	Specifies an IPv6 Network number to announce via BGP.
Step 7	**neighbor** *ipv6 address* **remote-as** *as-number*	Defines a BGP session between PE and CE routers.
Step 8	**neighbor** *address* **activate**	Activates the advertisement of the address family.
Step 9	**end**	Returns to privileged EXEC mode.
Step 10	**show bgp vrf** *vrf-name*	Verifies BGP configuration on the VRF.
Step 11	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Use the **no router bgp** *autonomous-system-number* global configuration command to delete the BGP routing process. Use the command with keywords to delete routing characteristics.

Multi-VRF CE Configuration Example

Figure 1-2 is a simplified example of the physical connections in a network similar to that in Figure 1-6. 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 1-2 Multi-VRF CE Configuration Example



Configuring Switch A

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-af)# 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 Ehernet 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 VRF11 between the CE and the PE. VLAN 20 is used by VRF12 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 vlan 10
Switch(config-if)# vrf forwarding v11
Switch(config-if)# ipv6 address 1000::1/64
Switch(config-if)# exit

Switch(config)# interface vlan 20
Switch(config-if)# vrf forwarding v12
Switch(config-if)# ipv6 address 2000::1/64
Switch(config-if)# exit

Switch(config)# interface vlan 208
Switch(config-if)# vrf forwarding v11
Switch(config-if)# ipv6 address 3000::1/64
Switch(config-if)# exit

Switch(config)# interface vlan 118
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)#r outer-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-router)# 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
```

Configuring Switch D

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
```

Configuring Switch E

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
```

Configuring the PE Switch B

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 1-3 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 1-4 Commands for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Displays a summary of access lists.
show ipv6 cef	Displays Cisco Express Forwarding for IPv6.

Table 1-4 Commands for Monitoring IPv6

Command	Purpose
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 ospf	Display IPv6 OSPF information.
show ipv6 prefix-list	Displays a list of IPv6 prefix lists.
show ipv6 protocols	Displays IPv6 routing protocols on the switch.
show ipv6 rip	Displays IPv6 RIP routing protocol status.
show ipv6 route	Displays the IPv6 route table entries.
show ipv6 routers	Displays the local IPv6 routers.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.

Table 1-5 Commands for Displaying EIGRP IPv6 Information

Command	Purpose
show ipv6 eigrp [*as-number*] *interface*	Displays information about interfaces configured for EIGRP IPv6.
show ipv6 eigrp [*as-number*] *neighbor*	Displays the neighbors discovered by EIGRP IPv6.
show ipv6 eigrp [*as-number*] *traffic*	Displays the number of EIGRP IPv6 packets sent and received.
show ipv6 eigrp topology [*as-number* | *ipv6-address*] [**active** | **all-links** | **detail-links** | **pending** | **summary** | **zero-successors**]	Displays EIGRP entries in the IPv6 topology table.

Table 1-6 Commands for Displaying IPv4 and IPv6 Address Types

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

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

```
Switch# show ipv6 interface
Vlan1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
```

```
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```