Configuring OSPFv3

This chapter contains the following sections:

Finding Feature Information

Your software release might not support all the features documented in this module. For the latest caveats and feature information, see the Bug Search Tool at https://tools.cisco.com/bugsearch/ and the release notes for your software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "New and Changed Information"chapter or the Feature History table in this chapter.

Information About OSPFv3

OSPFv3 is an IETF link-state protocol. An OSPFv3 router sends a special message, called a hello packet, out each OSPF-enabled interface to discover other OSPFv3 neighbor routers. Once a neighbor is discovered, the two routers compare information in the Hello packet to determine if the routers have compatible configurations. The neighbor routers attempt to establish adjacency, which means that the routers synchronize their link-state databases to ensure that they have identical OSPFv3 routing information. Adjacent routers share link-state advertisements (LSAs) that include information about the operational state of each link, the cost of the link, and any other neighbor information. The routers then flood these received LSAs out every OSPF-enabled interface so that all OSPFv3 routers eventually have identical link-state databases. When all OSPFv3 routers have identical link-state databases, the network is converged. Each router then uses Dijkstra's Shortest Path First (SPF) algorithm to build its route table.

You can divide OSPFv3 networks into areas. Routers send most LSAs only within one area, which reduces the CPU and memory requirements for an OSPF-enabled router.

OSPFv3 supports IPv6.

Comparison of OSPFv3 and OSPFv2

Much of the OSPFv3 protocol is the same as in OSPFv2. OSPFv3 is described in RFC 2740.

The key differences between the OSPFv3 and OSPFv2 protocols are as follows:

- OSPFv3 expands on OSPFv2 to provide support for IPv6 routing prefixes and the larger size of IPv6 addresses.
- LSAs in OSPFv3 are expressed as prefix and prefix length instead of address and mask.
- The router ID and area ID are 32-bit numbers with no relationship to IPv6 addresses.
- OSPFv3 uses link-local IPv6 addresses for neighbor discovery and other features.
- OSPFv3 can use the IPv6 authentication trailer (RFC 6506) or IPSec (RFC 4552) for authentication. However, neither of these options is supported on Cisco NX-OS.
- OSPFv3 redefines LSA types.

Hello Packet

OSPFv3 routers periodically send Hello packets on every OSPF-enabled interface. The hello interval determines how frequently the router sends these Hello packets and is configured per interface. OSPFv3 uses Hello packets for the following tasks:

- Neighbor discovery
- Keepalives
- Bidirectional communications
- Designated router election

The Hello packet contains information about the originating OSPFv3 interface and router, including the assigned OSPFv3 cost of the link, the hello interval, and optional capabilities of the originating router. An OSPFv3 interface that receives these Hello packets determines if the settings are compatible with the receiving interface settings. Compatible interfaces are considered neighbors and are added to the neighbor table.

Hello packets also include a list of router IDs for the routers that the originating interface has communicated with. If the receiving interface sees its own router ID in this list, then bidirectional communication has been established between the two interfaces.

OSPFv3 uses Hello packets as a keepalive message to determine if a neighbor is still communicating. If a router does not receive a Hello packet by the configured dead interval (usually a multiple of the hello interval), then the neighbor is removed from the local neighbor table.

Neighbors

An OSPFv3 interface must have a compatible configuration with a remote interface before the two can be considered neighbors. The two OSPFv3 interfaces must match the following criteria:

- Hello interval
- Dead interval
- Area ID
- Optional capabilities

If there is a match, the information is entered into the neighbor table:

- If there is a match, the information is entered into the neighbor table:
- Priority—Priority of the neighbor router. The priority is used for designated router election.
- State—Indication of whether the neighbor has just been heard from, is in the process of setting up bidirectional communications, is sharing the link-state information, or has achieved full adjacency.
- Dead time—Indication of how long since the last Hello packet was received from this neighbor.
- Link-local IPv6 Address—The link-local IPv6 address of the neighbor.
- Designated Router—Indication of whether the neighbor has been declared the designated router or backup designated router.
- Local interface—The local interface that received the Hello packet for this neighbor.

When the first Hello packet is received from a new neighbor, the neighbor is entered into the neighbor table in the initialization state. Once bidirectional communication is established, the neighbor state becomes two-way. ExStart and exchange states come next, as the two interfaces exchange their link-state database. Once this is all complete, the neighbor moves into the full state, which signifies full adjacency. If the neighbor fails to send any Hello packets in the dead interval, then the neighbor is moved to the down state and is no longer considered adjacent.

Adjacency

Not all neighbors establish adjacency. Depending on the network type and designated router establishment, some neighbors become fully adjacent and share LSAs with all their neighbors, while other neighbors do not.

Adjacency is established using Database Description packets, Link State Request packets, and Link State Update packets in OSPFv3. The Database Description packet includes the LSA headers from the link-state database of the neighbor. The local router compares these headers with its own link-state database and determines which LSAs are new or updated. The local router sends a Link State Request packet for each LSA that it needs new or updated information on. The neighbor responds with a Link State Update packet. This exchange continues until both routers have the same link-state information.

Designated Routers

Networks with multiple routers present a unique situation for OSPFv3. If every router floods the network with LSAs, the same link-state information is sent from multiple sources. Depending on the type of network, OSPFv3 might use a single router, the designated router (DR), to control the LSA floods and represent the network to the rest of the OSPFv3 area. If the DR fails, OSPFv3 uses the BDR.

Network types are as follows:

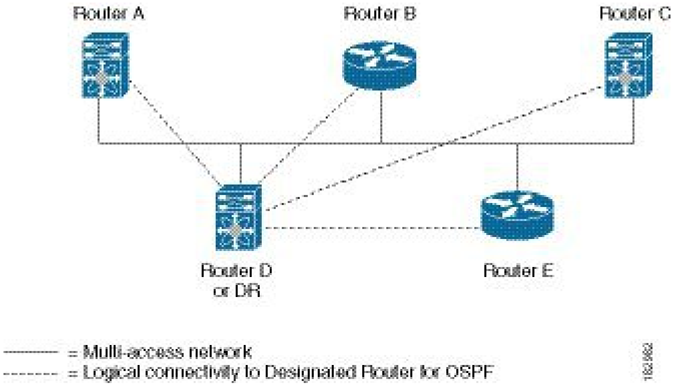
- Point-to-point—A network that exists only between two routers. All neighbors on a point-to-point network establish adjacency and there is no DR.
- Broadcast—A network with multiple routers that can communicate over a shared medium that allows broadcast traffic, such as Ethernet. OSPFv3 routers establish a DR and BDR that controls LSA flooding on the network. OSPFv3 uses the well-known IPv6 multicast addresses, FF02::5, and a MAC address of 0100.5300.0005 to communicate with neighbors.

The DR and BDR are selected based on the information in the Hello packet. When an interface sends a Hello packet, it sets the priority field and the DR and BDR field if it knows who the DR and BDR are. The routers follow an election procedure based on which routers declare themselves in the DR and BDR fields and the priority field in the Hello packet. As a final determinant, OSPFv3 chooses the highest router IDs as the DR and BDR.

All other routers establish adjacency with the DR and the BDR and use the IPv6 multicast address FF02::6 to send LSA updates to the DR and BDR. The Figure shows this adjacency relationship between all routers and the DR.

DRs are based on a router interface. A router might be the DR for one network and not for another network on a different interface.

Figure 1: DR in Multi-Access Network



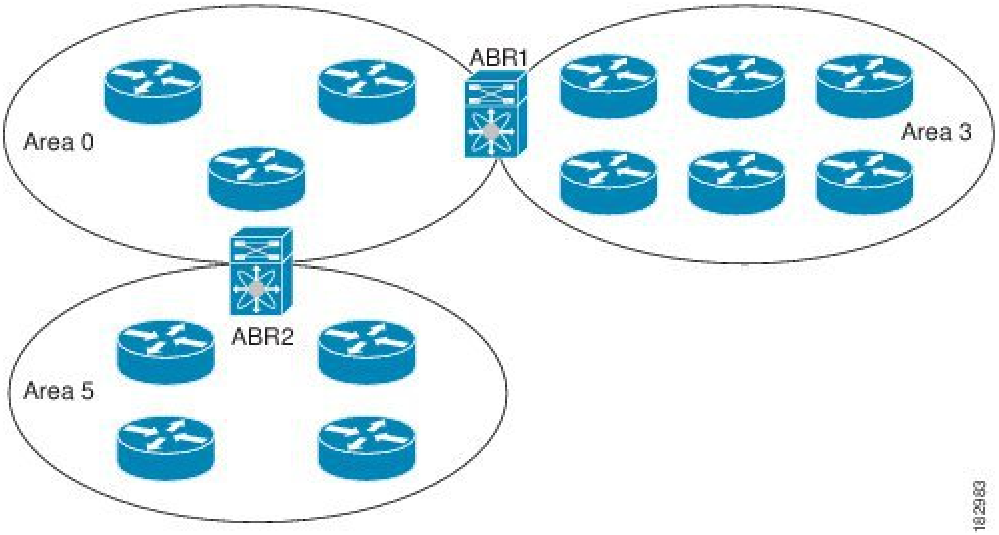
Areas

You can limit the CPU and memory requirements that OSPFv3 puts on the routers by dividing an OSPFv3 network into areas. An area is a logical division of routers and links within an OSPFv3 domain that creates separate subdomains. LSA flooding is contained within an area, and the link-state database is limited to links within the area. You can assign an area ID to the interfaces within the defined area. The Area ID is a 32-bit value that can be expressed as a number or in dotted decimal notation, such as 10.2.3.1.

Cisco NX-OS always displays the area in dotted decimal notation.

If you define more than one area in an OSPFv3 network, you must also define the backbone area, which has the reserved area ID of 0. If you have more than one area, then one or more routers become area border routers (ABRs). An ABR connects to both the backbone area and at least one other defined area.

Figure 2: OSPFv3 Areas



The ABR has a separate link-state database for each area which it connects to. The ABR sends Inter-Area Prefix (type 3) LSAs from one connected area to the backbone area. The backbone area sends summarized information about one area to another area. In the figure, Area 0 sends summarized information about Area 5 to Area 3.

OSPFv3 defines one other router type: the autonomous system boundary router (ASBR). This router connects an OSPFv3 area to another autonomous system. An autonomous system is a network controlled by a single technical administration entity. OSPFv3 can redistribute its routing information into another autonomous system or receive redistributed routes from another autonomous system.

Link-State Advertisement Types

OSPFv3 uses link-state advertisements (LSAs) to build its routing table.

	Names	Description
1	Router LSA	LSA sent by every router. This LSA includes the state and cost of all links but does not include prefix information. Router LSAs trigger an SPF recalculation. Router LSAs are flooded to the local OSPFv3 area.
2	Network LSA	LSA sent by the DR. This LSA lists all routers in the multi-access network but does not include prefix information. Network LSAs trigger an SPF recalculation.

	Names	Description
3	Inter-Area Prefix LSA	LSA sent by the area border router to an external area for each destination in local area. This LSA includes the link cost from the border router to the local destination.
4	Inter-Area Router LSA	LSA sent by the area border router to an external area. This LSA advertises the link cost to the ASBR only.
5	AS External LSA	LSA generated by the ASBR. This LSA includes the link cost to an external autonomous system destination. AS External LSAs are flooded throughout the autonomous system.
7	Type-7 LSA	LSA generated by the ASBR within an NSSA. This LSA includes the link cost to an external autonomous system destination. Type-7 LSAs are flooded only within the local NSSA.
8	Link LSA	LSA sent by every router, using a link-local flooding scope. This LSA includes the link-local address and IPv6 prefixes for this link.
9	Intra-Area Prefix LSA	LSA sent by every router. This LSA includes any prefix or link state changes. Intra-Area Prefix LSAs are flooded to the local OSPFv3 area. This LSA does not trigger an SPF recalculation.

	Names	Description
11	Grace LSAs	LSA sent by a restarting router, using a link-local flooding scope. This LSA is used for a graceful restart of OSPFv3.

Link Cost

Each OSPFv3 interface is assigned a link cost. The cost is an arbitrary number. By default, Cisco NX-OS assigns a cost that is the configured reference bandwidth divided by the interface bandwidth. By default, the reference bandwidth is 40 Gb/s. The link cost is carried in the LSA updates for each link.

Flooding and LSA Group Pacing

OSPFv3 floods LSA updates to different sections of the network, depending on the LSA type. OSPFv3 uses the following flooding scopes:

- Link-local—LSA is flooded only on the local link. Used for Link LSAs and Grace LSAs.
- Area-local—LSA is flooded throughout a single OSPF area only. Used for Router LSAs, Network LSAs, Inter-Area-Prefix LSAs, Inter-Area-Router LSAs, and Intra-Area-Prefix LSAs.
- AS scope—LSA is flooded throughout the routing domain. An AS scope is used for AS External LSAs.

LSA flooding guarantees that all routers in the network have identical routing information. LSA flooding depends on the OSPFv3 area configuration. The LSAs are flooded based on the link-state refresh time (every 30 minutes by default). Each LSA has its own link-state refresh time.

You can control the flooding rate of LSA updates in your network by using the LSA group pacing feature. LSA group pacing can reduce high CPU or buffer utilization. This feature groups LSAs with similar link-state refresh times to allow OSPFv3 to pack multiple LSAs into an OSPFv3 Update message.

By default, LSAs with link-state refresh times within 10 seconds of each other are grouped together. You should lower this value for large link-state databases or raise it for smaller databases to optimize the OSPFv3 load on your network.

Link-State Database

Each router maintains a link-state database for the OSPFv3 network. This database contains all the collected LSAs and includes information on all the routes through the network. OSPFv3 uses this information to calculate the bast path to each destination and populates the routing table with these best paths.

LSAs are removed from the link-state database if no LSA update has been received within a set interval, called the MaxAge. Routers flood a repeat of the LSA every 30 minutes to prevent accurate link-state information from being aged out. Cisco NX-OS supports the LSA grouping feature to prevent all LSAs from refreshing at the same time.

Multi-Area Adjacency

OSPFv3 multi-area adjacency allows you to configure a link on the primary interface that is in more than one area. This link becomes the preferred intra-area link in those areas. Multi-area adjacency establishes a point-to-point unnumbered link in an OSPFv3 area that provides a topological path for that area. The primary adjacency uses the link to advertise an unnumbered point-to-point link in the Router LSA for the corresponding area when the neighbor state is full.

The multi-area interface exists as a logical construct over an existing primary interface for OSPF; however, the neighbor state on the primary interface is independent of the multi-area interface. The multi-area interface establishes a neighbor relationship with the corresponding multi-area interface on the neighboring router.

OSPFv3 and the IPv6 Unicast RIB

OSPFv3 runs the Dijkstra shortest path first algorithm on the link-state database. This algorithm selects the best path to each destination based on the sum of all the link costs for each link in the path. The shortest path for each destination is then put in the OSPFv3 route table. When the OSPFv3 network is converged, this route table feeds into the IPv6 unicast RIB. OSPFv3 communicates with the IPv6 unicast RIB to do the following:

- Add or remove routes
- Handle route redistribution from other protocols
- Provide convergence updates to remove stale OSPFv3 routes and for stub router advertisements.

OSPFv3 also runs a modified Dijkstra algorithm for fast recalculation for Inter-Area Prefix, Inter-Area Router, AS-External, type-7, and Intra-Area Prefix (type 3, 4, 5, 7, 8) LSA changes.

Address Family Support

Cisco NX-OS supports multiple address families, such as unicast IPv6 and multicast IPv6. OSPFv3 features that are specific to an address family are as follows:

- Default routes
- Route summarization
- Route redistribution
- Filter lists for border routers
- SPF optimization

Use the **address-family ipv6 unicast** command to enter the IPv6 unicast address family configuration mode when configuring these features.

Authentication

You can configure authentication on OSPFv3 messages to prevent unauthorized or invalid routing updates in the network. OSPFv3 uses the Cisco NX-OS IPSecV6 secure sockets API to add authentication and encryption to its packets. It uses IPSec in transport mode with manually configured security association (SA) shared by all OSPFv3 routers in a link.

Cisco NX-OS OSPFv3 uses IPSec AH header with MD5 or SHA1 authentication. You can configure IPSec with a security policy, which is a combination of the security policy index (SPI) and a key.

OSPFv3 authentication can be configured at the following levels:

- Router / Process

- Area
- Interface

If you configure IPSec for an OSPFv3 area, the authentication is applied to all the interfaces in that area, except for the interfaces that have IPSec configured directly. If you configure IPSec for an OSPFv3 process, the authentication is applied on each interface in every area of that process. A security policy applied on an interface overrides the policy applied at the process or the area level.

Encryption

Beginning from Cisco Nexus Release 8.4.4, you can encrypt and authenticate OSPFv3 messages. OSPFv3 depends on IPSec for secure connection. IPSec supports two encapsulation types:

- Authentication Header (AH)
- Encapsulating Security Payload (ESP)

ESP configuration provides both encryption and authentication for OSPFv3 messages.

You can configure ESP at the following levels:

- Router
- Area
- Interface
- Virtual Links

Guidelines and Limitations for configuring ESP on OSPFv3

ESP configuration has the following guidelines and limitations:

- ESP configuration supports IPsec Transport Mode only.
- You can configure ESP on OSPFv3 for one SPI at one level, cannot configure two SPIs in one level.
- You cannot configure both encryption and authentication configurations for a same level.
- Supported encryption algorithms in ESP:
    - AES-CBC (128-bit)
    - 3DES-CBC
    - NULL
- Supported authentication algorithms in ESP:
    - SHA-1
    - NULL
- You cannot configure both ESP and AUTH algorithm as null in one ESP CLI.
- If ESP is not configured at local level, it inherits configuration from higher level, if configured:

  - If ESP is not configured at interface level, it inherits configuration from area level.
  - If ESP is not configured at area level, it inherits configuration from router level.
- On local level SPI, inherited data will be removed internally.

Note Ensure that the CoPP policy is customized to allow ESP packets, as default CoPP policy drops ESP packets.

Advanced Features

Cisco NX-OS supports advanced OSPFv3 features that enhance the usability and scalability of OSPFv3 in the network.

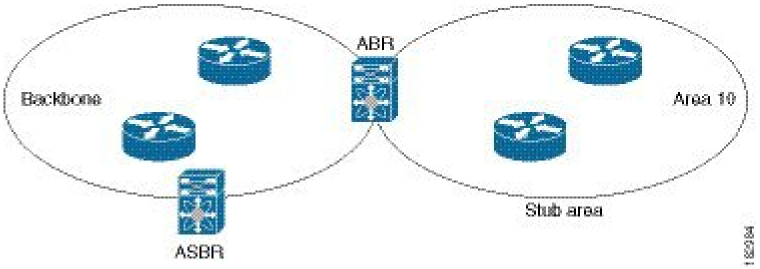
Stub Area

You can limit the amount of external routing information that floods an area by making it a stub area. A stub area is an area that does not allow AS External (type 5) LSAs. These LSAs are usually flooded throughout the local autonomous system to propagate external route information. Stub areas have the following requirements:

- All routers in the stub area are stub routers.
- No ASBR routers exist in the stub area.
- You cannot configure virtual links in the stub area.

The figure shows an example an OSPFv3 autonomous system where all routers in area 0.0.0.10 have to go through the ABR to reach external autonomous systems. Area 0.0.0.10 can be configured as a stub area.

Figure 3: Stub Area



Stub areas use a default route for all traffic that needs to go through the backbone area to the external autonomous system. The default route is an Inter-Area-Prefix LSA with the prefix length set to 0 for IPv6.

Not-So-Stubby Area

A Not-So-Stubby Area (NSSA) is similar to the stub area, except that an NSSA allows you to import autonomous system external routes within an NSSA using redistribution. The NSSA ASBR redistributes these routes and generates type-7 LSAs that it floods throughout the NSSA. You can optionally configure the ABR that connects the NSSA to other areas to translate this type-7 LSA to AS External (type 5) LSAs. The ABR then floods these

AS External LSAs throughout the OSPFv3 autonomous system. Summarization and filtering are supported during the translation.

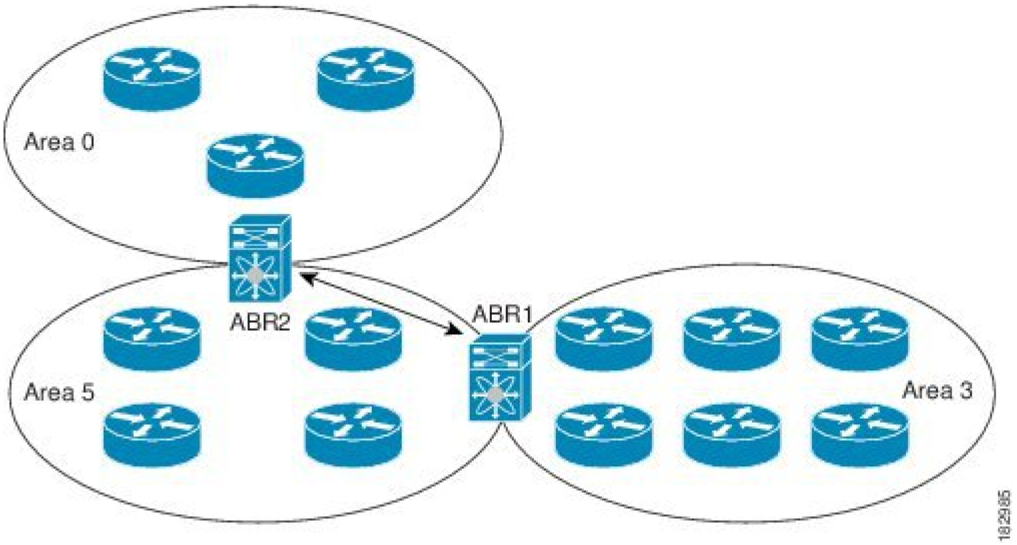
You can, for example, use NSSA to simplify administration if you are connecting a central site using OSPFv3 to a remote site that is using a different routing protocol. Before NSSA, the connection between the corporate site border router and a remote router could not be run as an OSPFv3 stub area because routes for the remote site could not be redistributed into a stub area. With NSSA, you can extend OSPFv3 to cover the remote connection by defining the area between the corporate router and remote router as an NSSA.

The backbone Area 0 cannot be an NSSA

Virtual Links

Virtual links allow you to connect an OSPFv3 area ABR to a backbone area ABR when a direct physical connection is not available. The figure shows a virtual link that connects Area 3 to the backbone area through Area 5.

Figure 4: Virtual Links



You can also use virtual links to temporarily recover from a partitioned area, which occurs when a link within the area fails, isolating part of the area from reaching the designated ABR to the backbone area.

Route Redistribution

OSPFv3 can learn routes from other routing protocols by using route redistribution. You configure OSPFv3 to assign a link cost for these redistributed routes or a default link cost for all redistributed routes.

Route redistribution uses route maps to control which external routes are redistributed. You must configure a route map with the redistribution to control which routes are passed into OSPFv3. A route map allows you to filter routes based on attributes such as the destination, origination protocol, route type, route tag, and so on. You can use route maps to modify parameters in the AS External (type 5) and NSSA External (type 7) LSAs before these external routes are advertised in the local OSPFv3 autonomous system.

OSPFv3 sets the type-5 LSA's forwarding address as described below:

- If the next-hop for the route is an attached-route then the forwarding address is the next-hop address for that route.
- If the next-hop for the route is a recursive route and next-hop's next-hop is an attached route then the forwarding address is the next-hop's next-hop address.

Route Summarization

Because OSPFv3 shares all learned routes with every OSPF-enabled router, you might want to use route summarization to reduce the number of unique routes that are flooded to every OSPF-enabled router. Route summarization simplifies route tables by replacing more-specific addresses with an address that represents all the specific addresses. For example, you can replace 2010:11:22:0:1000::1 and 2010:11:22:0:2000:679:1 with one summary address, 2010:11:22::/32.

Typically, you would summarize at the boundaries of area border routers (ABRs). Although you could configure summarization between any two areas, it is better to summarize in the direction of the backbone so that the backbone receives all the aggregate addresses and injects them, already summarized, into other areas. The two types of summarization are as follows:

- Inter-area route summarization
- External route summarization

You configure inter-area route summarization on ABRs, summarizing routes between areas in the autonomous system. To take advantage of summarization, assign network numbers in areas in a contiguous way to be able to lump these addresses into one range.

External route summarization is specific to external routes that are injected into OSPFv3 using route redistribution. You should make sure that external ranges that are being summarized are contiguous. Summarizing overlapping ranges from two different routers could cause packets to be sent to the wrong destination. Configure external route summarization on ASBRs that are redistributing routes into OSPF.

When you configure a summary address, Cisco NX-OS automatically configures a discard route for the summary address to prevent routing black holes and route loops.

High Availability and Graceful Restart

Cisco NX-OS provides a multilevel high-availability architecture. OSPFv3 supports stateful restart, which is also referred to as non-stop routing (NSR). If OSPFv3 experiences problems, it attempts to restart from its previous run-time state. The neighbors do not register any neighbor event in this case. If the first restart is not successful and another problem occurs, OSPFv3 attempts a graceful restart.

A graceful restart, or non-stop forwarding (NSF), allows OSPFv3 to remain in the data forwarding path through a process restart. When OSPFv3 needs to perform a graceful restart, it sends a link-local Grace (type 11) LSA. This restarting OSPFv3 platform is called NSF capable.

The Grace LSA includes a grace period, which is a specified time that the neighbor OSPFv3 interfaces hold onto the LSAs from the restarting OSPFv3 interface. (Typically, OSPFv3 tears down the adjacency and discards all LSAs from a down or restarting OSPFv3 interface.) The participating neighbors, which are called NSF helpers, keep all LSAs that originate from the restarting OSPFv3 interface as if the interface was still adjacent.

When the restarting OSPFv3 interface is operational again, it rediscovers its neighbors, establishes adjacency, and starts sending its LSA updates again. At this point, the NSF helpers recognize that the graceful restart has finished.

Stateful restart is used in the following scenarios:

- First recovery attempt after the process experiences problems
- ISSU
- User-initiated switchover using the **system switchover** command

Graceful restart is used in the following scenarios:

- Second recovery attempt after the process experiences problems within a 4-minute interval
- Manual restart of the process using the **restart ospfv3** command
- Active supervisor removal
- Active supervisor reload using the **reload module** *active-sup*command

Multiple OSPFv3 Instances

Cisco NX-OS supports multiple instances of the OSPFv3 protocol. By default, every instance uses the same system router ID. You must manually configure the router ID for each instance if the instances are in the same OSPFv3 autonomous system.

The OSPFv3 header includes an instance ID field to identify that OSPFv3 packet for a particular OSPFv3 instance. You can assign the OSPFv3 instance. The interface drops all OSPFv3 packets that do not have a matching OSPFv3 instance ID in the packet header.

Cisco NX-OS allows only one OSPFv3 instance on an interface.

SPF Optimization

Cisco NX-OS optimizes the SPF algorithm in the following ways:

- Partial SPF for Network (type 2) LSAs, Inter-Area Prefix (type 3) LSAs, and AS External (type 5) LSAs—When there is a change on any of these LSAs, Cisco NX-OS performs a faster partial calculation rather than running the whole SPF calculation.
- SPF timers—You can configure different timers for controlling SPF calculations. These timers include exponential backoff for subsequent SPF calculations. The exponential backoff limits the CPU load of multiple SPF calculations.

Virtualization Support

OSPFv3 supports virtual routing and forwarding (VRF) instances. VRFs exist within virtual device contexts (VDCs). By default, Cisco NX-OS places you in the default VDC and default VRF unless you specifically configure another VDC and VRF. Each OSPFv3 instance can support multiple VRFs, up to the system limit. For more information, see the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guid*e.

Prerequisites for OSPFv3

OSPFv3 has the following prerequisites:

- You must be familiar with routing fundamentals to configure OSPFv3.
- You must be logged on to the switch.
- You have configured at least one interface for IPv6 that is capable of communicating with a remote OSPFv3 neighbor.
- You have completed the OSPFv3 network strategy and planning for your network. For example, you must decide whether multiple areas are required.
- You have enabled OSPF.
- You are familiar with IPv6 addressing and basic configuration.

Guidelines and Limitations for OSPFv3

OSPFv3 has the following configuration guidelines and limitations:

- You can have up to four instances of OSPFv3 in a VDC.
- Before Cisco NX-OS Release 6.2(2), Bidirectional Forwarding Detection (BFD) was not supported for OSPFv3. In Cisco NX-OS Release 6.2(2) and later releases, BFD includes a client for OSPFv3.
- Cisco NX-OS displays areas in dotted decimal notation regardless of whether you enter the area in decimal or dotted decimal notation.
- MTU configured at interface level works in either the data plane or in the control plane but not at both planes at the same time.

  When you configure MTU with a size lower than the supported size in data and control planes a few features that have minimum MTU requirements may not work in both the planes.

  For example, MPLS VPN is supported in the data plane since this plane supports the MTU of 1500 bytes that the MPLS VPN requires. But control plane does not support MPLS VPN because this plane cannot handle the 1500-byte packets.

  To make the configured MTU work in control plane for MPLS VPN, you need to manually configure the OSPF packet size (by using the **packet-size***size* command) so that OSPF works on the control plane. This is applicable from Cisco NX-OS Release 8.3(2) onwards.

  The **packet-size***size* command is supported on the Ethernet, SVI, and GRE tunnel interfaces.

- If you configure OSPFv3 in a virtual port channel (vPC) environment, use the following timer commands in router configuration mode on the core switch to ensure fast OSPF convergence when a vPC peer link is shut down:

  ```
  switch (config-router)# timers throttle spf 1 50 50
  switch (config-router)# timers lsa-arrival 10
  ```

- The value of object OSPFv3 router ID differs from RFC 5643 for traps ospfv3NbrRestartHelperStatusChange and ospfv3VirtNbrRestartHelperStatusChange. As per the RFC 5643, the value of object OSPFv3 router ID should be the router ID of the originator of the trap. But the current implementation will provide the router ID of the neighbor for both ospfv3NbrRestartHelperStatusChange and ospfv3VirtNbrRestartHelperStatusChange.
- Only the first four OSPFv3 instances are supported with MPLS LDP and MPLS TE.

- In scaled scenarios, when the number of interfaces and link-state advertisements in an OSPFv3 process is large, the snmp-walk on OSPF MIB objects is expected to time out with a small-value timeout at the SNMP agent. If you observe a timeout on the querying SNMP agent while polling OSPF MIB objects, increase the timeout value on the polling SNMP agent.
- If there is a particular OSPFv3 prefix that is learnt through type-5 as well as type-7, and both have different forwarding addresses, then these two route types are not comparable as per RFC3101, Section 2.5, step 6(e). (This applies only if the same destination/cost/non-zero forwarding addresses are there). OSPF will therefore do ECMP with all available next-hops.
- NXOS OSPF and U6RIB store only one route-type per route. If there is a mix of route-type across next-hops, only one of them, (the new path type) will be shown for all next hops.

  Currently, route-type is a route property, and not a next-hop property.
- The **default-information-originate always** command advertises the OSPF defaut route from Cisco NX-OS Release 7.3(5)D1(1) and later releases and from Cisco NX-OS Release 8.0(1) and later releases in 8.x release train.
- The following guidelines and limitations apply to the administrative distance feature, which is supported beginning with Cisco NX-OS Release 6.1:
  - When an OSPF route has two or more equal cost paths, configuring the administrative distance is non-deterministic for the **match ip route-source** command.
  - For matching route sources in OSPFv3 routes, you must configure **match ip route-source** instead of **match ipv6 route-source** because the route sources and router IDs for OSPFv3 are IPv4 addresses.
  - Configuring the administrative distance is supported only for the **match route-type**, **match ipv6 address prefix-list**, and **match ip route-source prefix-list** commands. The other match statements are ignored.
  - The discard route is always assigned an administrative distance of 220. No configuration in the table map applies to OSPF discard routes.
  - There is no preference among the **match route-type**, **match ipv6 address**, and **match ip route-source** commands for setting the administrative distance of OSPF routes. In this way, the behavior of the table map for setting the administrative distance in Cisco NX-OS OSPF is different from that in Cisco IOS OSPF.
  - In Cisco NX-OS Release 6.2(6a) and later releases, you can filter next-hop paths for an OSPF route to prevent the path from being added to the RIB. Before Cisco NX-OS Release 6.2(6a), filtering on a specific path was ignored and the entire route was not added to the RIB.
- If you are familiar with the Cisco IOS CLI, be aware that the Cisco NX-OS commands for this feature might differ from the Cisco IOS commands that you would use.

Default Settings for OSPFv3

Table 1: Default OSPFv3 Parameters

Parameters	Default
Administrative distance	110
Hello interval	10 seconds
Dead interval	40 seconds
Discard routes	Enabled
Graceful restart grace period	60 seconds
Graceful restart notify period	15 seconds
OSPFv3 feature	Disabled
Stub router advertisement announce time	600 seconds
Reference bandwidth for link cost calculation	40 Gb/s
LSA minimal arrival time	1000 milliseconds
LSA group pacing	10 seconds
SPF calculation initial delay time	0 milliseconds
SPF calculation hold time	5000 milliseconds
SPF calculation initial delay time	0 milliseconds

Configuring Basic OSPFv3

Configure OSPFv3 after you have designed your OSPFv3 network.

Enabling OSPFv3

Before you begin

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# [**no**] **feature ospfv3**	Enables OSPFv3. To disable the OSPFv3 feature and remove all associated configurations, use the **no** form of the command.
Step 3	(Optional) switch(config)# **show feature**	Displays enabled and disabled features.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Creating an OSPFv3 Instance

The first step in configuring OSPFv3 is to create an instance or OSPFv3 instance. You assign a unique instance tag for this OSPFv3 instance. The instance tag can be any string. For each OSPFv3 instance, you can also configure the following optional parameters:

- Router ID—Configures the router ID for this OSPFv3 instance. If you do not use this parameter, the router ID selection algorithm is used.
- Administrative distance—Rates the trustworthiness of a routing information source.
- Log adjacency changes—Creates a system message whenever an OSPFv3 neighbor changes its state.
- Name lookup—Translates OSPF router IDs to host names, either by looking up the local hosts

  database or querying DNS names in IPv6.
- Maximum paths—Sets the maximum number of equal paths that OSPFv3 installs in the route table for a particular destination. Use this parameter for load balancing between multiple paths.
- Reference bandwidth—Controls the calculated OSPFv3 cost metric for a network. The calculated cost is the reference bandwidth divided by the interface bandwidth. You can override the calculated cost by assigning a link cost when a network is added to the OSPFv3 instance.

Note The OSPF router ID changes without a restart on a Cisco Nexus 7000 series switch when you have not configured a manual router ID in the following cases:

- Configuring an SVI or physical interface with a higher IP address than the current router ID on a setup without any configured loopback interfaces.
- Configuring a loopback interface with any given IP address on a setup without any previously configured loopback interfaces.
- Configuring a loopback interface with a higher IP address than the IP address of an existing configured loopback interface.

When a router ID changes, OSPF has to re-advertise all LSAs with the new router ID. To avoid this issue, you need to configure a manual OSPF router ID.

Before you begin

You must enable OSPFv3.

Ensure that the OSPFv3 instance tag that you plan on using is not already in use on this router.

Use the **show ospfv3** *instance-tag* command to verify that the instance tag is not in use.

OSPFv3 must be able to obtain a router identifier (for example, a configured loopback address) or you must configure the router ID option.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag. **Note** The **no router ospfv3** *instance tag* command does not remove OSPF configuration in interface mode. You must manually remove any OSPFv3 commands configured in interface mode.
Step 3	(Optional) switch(config-router)# **router-id** *ip-address*	Configures the OSPFv3 router ID. This ID uses the dotted decimal notation and identifies this OSPFv3 instance and must exist on a configured interface in the system. This command restarts the OSPF process automatically and changes the router id after it is configured.
Step 4	(Optional) switch(config-router)# **show ipv6 ospfv3** *instance-tag*	Displays OSPFv3 information.
Step 5	(Optional) switch(config-router)# **log-adjacency-changes** [**detail**]	Generates a system message whenever a neighbor changes state.
Step 6	(Optional) switch(config-router)# **passive-interface default**	Suppresses routing updates on all interfaces. This command is overridden by the VRF or interface command mode configuration.
Step 7	(Optional) switch(config-router-af)# **distance** *numbers*	Configures the administrative distance for this OSPFv3 instance. The range is from 1 to 255. The default is 110.
Step 8	switch(config-router-af)# **maximum-paths** *paths*	Configures the maximum number of equal OSPFv3 paths to a destination in the route table. The range is from 1 to 16. The default is 8. This command is used for load balancing.

	Command or Action	Purpose
Step 9	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create an OSPFv3 instance:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# copy running-config startup-config
```

Configuring OSPFv3 Packet Size

MTU configured at interface level works in either the data plane or in the control plane but not at both planes at the same time.

When you configure MTU with a size lower than the supported size in data and control planes a few features that have minimum MTU requirements may not work in both the planes.

For example, MPLS VPN is supported in the data plane since this plane supports the MTU of 1500 bytes that the MPLS VPN requires. But control plane does not support MPLS VPN because this plane cannot handle the 1500-byte packets.

To make the configured MTU work in control plane for MPLS VPN, you need to manually configure the OSPF packet size so that OSPF works on the control plane. This is applicable from Cisco NX-OS Release 8.3(2) onwards.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag. **Note** The **no router ospfv3** *instance-tag* command does not remove OSPF configuration in interface mode. You must manually remove any OSPFv3 commands configured in interface mode.
Step 3	switch(config-router)# **router-id** *ip-address*	Configures the OSPFv3 router ID. This ID uses the dotted decimal notation and identifies this OSPFv3 instance and must exist on a configured interface in the system.

	Command or Action	Purpose
		This command restarts the OSPF process automatically and changes the router id after it is configured.
Step 4	switch(config-router)# **ospfv3 packet-size** *size*	• Configures the OSPFv3 packet size. The size range is from 1280 to 9212 bytes. • You can configure the packet-size in the interface configuration mode also. • You can configure the **packet-size** *size* command even if the**ip ospf mtu-ignore** command is already configured in the network.
Step 5	(Optional) switch(config-router)# **show ospfv3 interface**	Displays OSPF information.

Example

This example shows how to configure the OSPFv3 packet-size:

```
router ospf 1
  router-id 3.3.3.3
  [no] packet-size 2000
```

This example shows the display of the configured OSPFv3 packet-size:

```
Switch (config-router)# show ospfv3 interface ethernet 1/25
Ethernet1/25 is up, line protocol is up
    IP address 1.0.0.1/24
    ---------    snip ---------------
    Number of opaque link LSAs: 0, checksum sum 0
    Max Packet Size: 2000
```

Configuring Networks in OSPFv3

You can configure a network to OSPFv3 by associating it through the interface that the router uses to connect to that network. You can add all networks to the default backbone area (Area 0), or you can create new areas using any decimal number or an IP address.

Note All areas must connect to the backbone area either directly or through a virtual link.

Note OSPFv3 is not enabled on an interface until you configure a valid IPv6 address for that interface.

Before you begin

You must enable OSPFv3.

Ensure that you are in the correct VDC (or use the switchto vdc command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *interface-type slot/port*	Enters interface configuration mode.
Step 3	switch(config-if)# **ipv6 address** *ipv6-prefix/length*	Assigns an IPv6 address to this interface.
Step 4	switch(config-if)# **ipv6 router ospfv3** *instance-tag* **area** *area-id* [**secondaries none**]	Adds the interface to the OSPFv3 instance and area.
Step 5	(Optional) switch(config-if)# **show ipv6 ospfv3** *instance-tag* **interface** *interface-type slot/port*	Displays OSPFv3 information.
Step 6	(Optional) switch(config-if)# **ospfv3 cost** *number*	Configures the OSPFv3 cost metric for this interface. The default is to calculate a cost metric, based on the reference bandwidth and interface bandwidth. The range is from 1 to 65535.
Step 7	(Optional) switch(config-if)# **ospfv3 dead-interval** *seconds*	Configures the OSPFv3 dead interval, in seconds. The range is from 1 to 65535. The default is four times the hello interval, in seconds.
Step 8	(Optional) switch(config-if)# **ospfv3 hello-interval** *seconds*	Configures the OSPFv3 hello interval, in seconds. The range is from 1 to 65535. The default is 10 seconds.
Step 9	(Optional) switch(config-if)# **ospfv3 instance** *instance*	Configures the OSPFv3 instance ID. The range is from 0 to 255. The default is 0. The instance ID is link-local in scope.
Step 10	(Optional) switch(config-if)# **ospfv3 mtu-ignore**	Configures OSPFv3 to ignore any IP maximum transmission unit (MTU) mismatch with a neighbor. The default is to not establish adjacency if the neighbor MTU does not match the local interface MTU.
Step 11	(Optional) switch(config-if)# **ospfv3 network** {**broadcast | point-point**}	Sets the OSPFv3 network type.
Step 12	(Optional) switch(config-if)# **ospfv3 priority** *number*	Configures the OSPFv3 priority, used to determine the DR for an area. The range is from 0 to 255. The default is 1.

	Command or Action	Purpose
Step 13	(Optional) switch(config-if)# **ospfv3 shutdown**	Shuts down the OSPFv3 instance on this interface.
Step 14	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to add a network area 0.0.0.10 in OSPFv3 instance 201:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# ipv6 address 2001:0DB8::1/48
switch(config-if)# ipv6 ospfv3 201 area 0.0.0.10
switch(config-if)# copy running-config startup-config
```

Configuring Advanced OSPFv3

Configure OSPFv3 after you have designed your OSPFv3 network.

Configuring Filter Lists for Border Routers

You can separate your OSPFv3 domain into a series of areas that contain related networks. All areas must connect to the backbone area through an area border router (ABR). OSPFv3 domains can connect to external domains as well through an autonomous system border router (ASBR).

ABRs have the following optional configuration parameters:

- Area range—Configures route summarization between areas.
- Filter list—Filters the Inter-Area Prefix (type 3) LSAs on an ABR that are allowed in from an external area.

ASBRs also support filter lists.

Before you begin

Create the route map that the filter list uses to filter IP prefixes in incoming or outgoing Inter-Area Prefix (type 3) LSAs.

You must enable OSPFv3.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag
Step 3	switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 4	switch(config-router-af)# **area** *area-id* **filter-list route-map** *map-name* {**in** | **out**}	Filters incoming or outgoing Inter-Area Prefix (type 3) LSAs on an ABR.
Step 5	(Optional) switch(config-if)# **show ipv6 ospfv3 policy statistics area** *id* **filter-list** {**in** | **out**}	Displays OSPFv3 policy information.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to enable graceful restart if it has been disabled:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# area 0.0.0.10 filter-list route-map FilterLSAs in
switch(config-router-af)# copy running-config startup-config
```

Configuring Stub Areas

You can configure a stub area for part of an OSPFv3 domain where external traffic is not necessary. Stub areas block AS External (type 5) LSAs, limiting unnecessary routing to and from selected networks. You can optionally block all summary routes from going into the stub area.

Before you begin

You must enable OSPF.

Ensure that there are no virtual links or ASBRs in the proposed stub area.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **area** *area-id* **stub**	Creates this area as a stub area.

	Command or Action	Purpose
Step 4	(Optional) switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 5	(Optional) switch(config-router-af)# **area** *area-id* **default cost** *cost*	Sets the cost metric for the default summary route sent into this stub area. The range is from 0 to 16777215.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This shows how to create a stub area that blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 stub no-summary
switch(config-router)# copy running-config startup-config
```

Configuring a Totally Stubby Area

You can create a totally stubby area and prevent all summary route updates from going into the stub area.

To create a totally stubby area, use the following command in router configuration mode:

Procedure

	Command or Action	Purpose
Step 1	switch(config-router)# **area** *area-id* **stub no-summary**	Creates this area as a totally stubby area.

Configuring NSSA

You can configure an NSSA for part of an OSPFv3 domain where limited external traffic is required. You can optionally translate this external traffic to an AS External (type 5) LSA and flood the OSPFv3 domain with this routing information. An NSSA can be configured with the following optional parameters:

- No redistribution—Redistributes routes that bypass the NSSA to other areas in the OSPFv3 autonomous system. Use this option when the NSSA ASBR is also an ABR.
- Default information originate—Generates a Type-7 LSA for a default route to the external autonomous system. Use this option on an NSSA ASBR if the ASBR contains the default route in the routing table. This option can be used on an NSSA ABR whether or not the ABR contains the default route in the routing table.
- Route map—Filters the external routes so that only those routes you want are flooded throughout the NSSA and other areas.

- Translate—Translates Type-7 LSAs to AS External (type 5) LSAs for areas outside the NSSA. Use this command on an NSSA ABR to flood the redistributed routes throughout the OSPFv3 autonomous system. You can optionally suppress the forwarding address in these AS External LSAs.
- No summary—Blocks all summary routes from flooding the NSSA. Use this option on the NSSA ABR.

Before you begin

You must enable OSPF.

Ensure that there are no virtual links in the proposed NSSA and that it is not the backbone area.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**route-map** *map-name*] [**no-summary**] [**translate type7** {**always** | **never**} [**suppress-fa**]]	Creates this area as an NSSA.
Step 4	(Optional) switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 5	(Optional) switch(config-router-af)# **area** *area-id* **default cost** *cost*	Sets the cost metric for the default summary route sent into this NSSA. The range is from 0 to 16777215.
Step 6	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create an NSSA that blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 nssa no-summary
switch(config-router)# copy running-config startup-config
```

This example shows how to create an NSSA that generates a default route:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 nssa default-info-originate
switch(config-router)# copy running-config startup-config
```

This example shows how to create an NSSA that filters external routes and blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 nssa route-map ExternalFilter no-summary
switch(config-router)# copy running-config startup-config
```

This example shows how to create an NSSA that always translates Type-7 LSAs to AS External (type 5) LSAs:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 nssa translate type 7 always
switch(config-router)# copy running-config startup-config
```

This example shows how to create an NSSA that blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 nssa no-summary
switch(config-router)# copy running-config startup-config
```

Configuring Multi-Area Adjacency

You can add more than one area to an existing OSPFv3 interface. The additional logical interfaces support multi-area adjacency.

Before you begin

You must enable OSPF.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Ensure that you have configured a primary area for the interface.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** *interface-type slot/port*	Enters interface configuration mode.
Step 3	switch(config-if)# **ipv6 router ospfv3** *instance-tag* **multi-area** *area-id*	Adds the interface to another area.
Step 4	(Optional) switch(config-if)# **show ipv6 ospfv3** *instance-tag* **interface** *interface-type slot/port*	Displays OSPFv3 information.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to add a second area to an OSPFv3 interface:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# ipv6 address 2001:0DB8::1/48
switch(config-if)# ipv6 ospfv3 201 area 0.0.0.10
switch(config-if)# ipv6 ospfv3 201 multi-area 20
switch(config-if)# copy running-config startup-config
```

Configuring Virtual Links

A virtual link connects an isolated area to the backbone area through an intermediate area. You can configure the following optional parameters for a virtual link:

- Dead interval—Sets the time that a neighbor waits for a Hello packet before declaring the local router as dead and tearing down adjacencies.
- Hello interval—Sets the time between successive Hello packets.
- Retransmit interval—Sets the estimated time between successive LSAs.
- Transmit delay—Sets the estimated time to transmit an LSA to a neighbor.

Note You must configure the virtual link on both routers involved before the link becomes active.

Before you begin

You must enable OSPF.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **area** *area-id* **virtual-link** *router-id*	Creates one end of a virtual link to a remote router. You must create the virtual link on that remote router to complete the link.
Step 4	(Optional) switch(config-if)# **show ipv6 ospfv3 virtual-link** [**brief**]	Displays OSPFv3 virtual link information.
Step 5	(Optional) switch(config-router-vlink)# **dead-interval** *seconds*	Configures the OSPFv3 dead interval, in seconds. The range is from 1 to 65535. The

	Command or Action	Purpose
		default is four times the hello interval, in seconds.
Step 6	(Optional) switch(config-router-vlink)# **hello-interval** *seconds*	Configures the OSPFv3 hello interval, in seconds. The range is from 1 to 65535. The default is 10 seconds.
Step 7	(Optional) switch(config-router-vlink)# **retransmit-interval** *seconds*	Configures the OSPFv3 retransmit interval, in seconds. The range is from 1 to 65535. The default is 5.
Step 8	(Optional) switch(config-router-vlink)# **transmit-delay** *seconds*	Configures the OSPFv3 transmit-delay, in seconds. The range is from 1 to 450. The default is 1.
Step 9	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

These examples show how to create a simple virtual link between two ABRs:

Configuration for ABR 1 (router ID 2001:0DB8::1) is as follows:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 virtual-link 2001:0DB8::10
switch(config-router)# copy running-config startup-config
```

Configuration for ABR 2 (router ID 2001:0DB8::10) is as follows:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# area 0.0.0.10 virtual-link 2001:0DB8::1
switch(config-router)# copy running-config startup-config
```

Configuring Redistribution

You can redistribute routes learned from other routing protocols into an OSPFv3 autonomous system through the ASBR.

You can configure the following optional parameters for route redistribution in OSPF:

- Default information originate—Generates an AS External (type 5) LSA for a default route to the external autonomous system.

Note Default information originate ignores **match** statements in the optional route map.

- Default metric—Sets all redistributed routes to the same cost metric.

Note If you redistribute static routes, Cisco NX-OS also redistributes the default static route.

Before you begin

Create the necessary route maps used for redistribution.

You must enable OSPF.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 4	switch(config-router-af)# **redistribute** {**bgp***id* | **direct** | **isis** *id* | **rip** *id* | **static**} **route-map** *map-name*	Redistributes the selected protocol into OSPFv3 through the configured route map. **Note** If you redistribute static routes, Cisco NX-OS also redistributes the default static route.
Step 5	switch(config-router-af)# **default-information originate** [**always**] [**route-map** *map-name*]	Creates a default route into this OSPFv3 domain if the default route exists in the RIB. Use the following optional keywords: • **always** —Always generates the default route of 0.0.0. even if the route does not exist in the RIB. • **route-map**—Generates the default route if the route map returns true. **Note** This command ignores **match** statements in the route map.
Step 6	switch(config-router-af)# **default-metric** *cost*	Sets the cost metric for the redistributed routes. The range is from 1 to 16777214. This command does not apply to directly connected routes. Use a route map to set the default metric for directly connected routes.

	Command or Action	Purpose
Step 7	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to redistribute the Border Gateway Protocol (BGP) into OSPFv3:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# redistribute bgp route-map FilterExternalBGP
switch(config-router-af)# copy running-config startup-config
```

Limiting the Number of Redistributed Routes

Route redistribution can add many routes to the OSPFv3 route table. You can configure a maximum limit to the number of routes accepted from external protocols. OSPFv3 provides the following options to configure redistributed route limits:

- Fixed limit—Logs a message when OSPFv3 reaches the configured maximum. OSPFv3 does not accept any more redistributed routes. You can optionally configure a threshold percentage of the maximum where OSPFv3 logs a warning when that threshold is passed.
- Warning only—Logs a warning only when OSPFv3 reaches the maximum. OSPFv3 continues to accept redistributed routes.
- Withdraw—Starts the configured timeout period when OSPFv3 reaches the maximum. After the timeout period, OSPFv3 requests all redistributed routes if the current number of redistributed routes is less than the maximum limit. If the current number of redistributed routes is at the maximum limit, OSPFv3 withdraws all redistributed routes. You must clear this condition before OSPFv3 accepts more redistributed routes. You can optionally configure the timeout period.

Before you begin

You must enable OSPF.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.

	Command or Action	Purpose
Step 4	switch(config-router)# **redistribute** {**bgp***id* | **direct** | **isis** *id* | **rip** *id* | **static**} **route-map** *map-name*	Redistributes the selected protocol into OSPFv3 through the configured route map.
Step 5	switch(config-router)# **redistribute maximum-prefix***max* [*threshold*] [**warning-only** | **withdraw** [*num-retries timemout*]]	Specifies a maximum number of prefixes that OSPFv2 distributes. The range is from 0 to 65536. Optionally, specifies the following: • *threshold*—Percent of maximum prefixes that triggers a warning message. • **warning-only**—Logs an warning message when the maximum number of prefixes is exceeded. • **withdraw**—Withdraws all redistributed routes and optionally tries to retrieve the redistributed routes. The num-retries range is from 1 to 12. The timeout range is from 60 to 600 seconds. The default is 300 seconds.
Step 6	(Optional) **show running-config ospfv3** **Example:** `switch(config-router)# show running-config ospf`	Displays the OSPFv3 configuration.
Step 7	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to limit the number of redistributed routes into OSPF:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# redistribute bgp route-map FilterExternalBGP
switch(config-router-af)# copy running-config startup-config
```

Configuring Route Summarization

You can configure route summarization for inter-area routes by configuring an address range that is summarized. You can also configure route summarization for external, redistributed routes by configuring a summary address for those routes on an ASBR.

Before you begin

You must enable OSPF.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 4	switch(config-router-af)# **area** *area-id* **range** *ipv6-prefix/length* [**no-advertise**] [**cost** *cost*]	Creates a summary address on an ABR for a range of addresses and optionally advertises this summary address in a Inter-Area Prefix (type 3) LSA. The cost range is from 0 to 16777215.
Step 5	switch(config-router-af)# **summary-address** *ipv6-prefix/length* [**no-advertise**] [**tag** *tag*]	Creates a summary address on an ASBR for a range of addresses and optionally assigns a tag for this summary address that can be used for redistribution with route maps.
Step 6	(Optional) switch(config-router)# **show ipv6 ospfv3 summary-address**	Displays information about OSPFv3 summary addresses.
Step 7	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This shows how to create a stub area that blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router)# area 0.0.0.10 range 2001:0DB8::/48
switch(config-router)# copy running-config startup-config
```

This example shows how to create summary addresses on an ASBR:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router)# summary-address 2001:0DB8::/48
switch(config-router)# copy running-config startup-config
```

Configuring the Administrative Distance of Routes

Beginning with Cisco NX-OS Release 6.1, you can set the administrative distance of routes added by OSPFv3 into the RIB.

The administrative distance is a rating of the trustworthiness of a routing information source. A higher value indicates a lower trust rating. Typically, a route can be learned through more than one routing protocol. The administrative distance is used to discriminate between routes learned from more than one routing protocol. The route with the lowest administrative distance is installed in the IP routing table.

Before you begin

Ensure that you have enabled OSPFv3.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

See the guidelines and limitations for this feature.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospf** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 4	switch(config-router-af)# [**no**] **table-map** *map-name* [**filter**]	Configures the policy for filtering or modifying OSPFv2 routes before sending them to the RIB. You can enter up to 63 alphanumeric characters for the map name. The **filter** keyword specifies that only routes that are permitted by the route map(*map-name*) configuration are downloaded to the routing information base (RIB).
Step 5	switch(config-router-af)# **exit**	Exits router address-family configuration mode.
Step 6	switch(config-router)# **exit**	Exits router configuration mode.
Step 7	switch(config)# **route-map** *map-name* [**permit** | **deny**] [*seq*]	Creates a route map or enters route-map configuration mode for an existing route map. Use *seq* to order the entries in a route map. **Note** The **permit** option enables you to set the distance. If you use the **deny** option, the default distance is applied.
Step 8	switch(config-route-map)# **match route-type** *route-type*	Matches against one of the following route types: • external—The external route (BGP, EIGRP, and OSPF type 1 or 2) • inter-area—OSPF inter-area route

	Command or Action	Purpose
		• internal—The internal route (including the OSPF intra- or inter-area) • intra-area—OSPF intra-area route • nssa-external—The NSSA external route (OSPF type 1 or 2) • type-1—The OSPF external type 1 route • type-2—The OSPF external type 2 route
Step 9	switch(config-route-map)# **match ip route-source prefix-list** *name*	Matches the IPv6 route source address or router ID of a route to one or more IP prefix lists. Use the **ip prefix-list** command to create the prefix list. **Note** For OSPFv3, the router ID is 4 bytes.
Step 10	switch(config-route-map)# **match ipv6 address prefix-list** *name*	Matches against one or more IPv6 prefix lists. Use the **ip prefix-list** command to create the prefix list.
Step 11	switch(config-route-map)# **set distance** *value*	Sets the administrative distance of routes for OSPFv3. The range is from 1 to 255.
Step 12	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure the OSPFv3 administrative distance for inter-area routes to 150, for external routes to 200, and for all prefixes in prefix list p1 to 190:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# address-family ipv6 unicast
switch(config-router-af)# table-map foo
switch(config-router)# exit
switch(config)# exit
switch(config)# route-map foo permit 10
switch(config-route-map)# match route-type inter-area
switch(config-route-map)# set distance 150
switch(config)# route-map foo permit 20
switch(config-route-map)# match route-type external
switch(config-route-map)# set distance 200
switch(config)# route-map foo permit 30
switch(config-route-map)# match ip route-source prefix-list p1
switch(config-route-map)# match ipv6 address prefix-list p1
switch(config-route-map)# set distance 190
```

The following example shows how to configure a route map for blocking the next hops that are learned through VLAN 10:

```
switch(config)# route-map Filter-OSPF 10 deny
switch(config-route-map)# match interface VLAN 10
switch(config-route-map)# exit
switch(config)# route-map Filter-OSPF 20 permit
```

The following example shows how to configure the **table-map** command with the **filter** keyword to use a route map (Filter-OSPF) to remove the next-hop path that is learned through VLAN 10 but not the next-hop path that is learned through VLAN 20:

```
switch(config)# route ospfv3 p1
switch(config-router)# table-map Filter-OSPF filter
```

Modifying the Default Timers

OSPFv3 includes a number of timers that control the behavior of protocol messages and shortest path first (SPF) calculations. OSPFv3 includes the following optional timer parameters:

- LSA arrival time—Sets the minimum interval allowed between LSAs arriving from a neighbor. LSAs that arrive faster than this time are dropped.
- Pacing LSAs—Sets the interval at which LSAs are collected into a group and refreshed, checksummed, or aged. This timer controls how frequently LSA updates occur and optimizes how many are sent in an LSA update message.
- Throttle LSAs—Sets rate limits for generating LSAs. This timer controls how frequently LSAs are generated after a topology change occurs.
- Throttle SPF calculation—Controls how frequently the SPF calculation is run.

At the interface level, you can also control the following timers:

- Retransmit interval—Sets the estimated time between successive LSAs
- Transmit delay—Sets the estimated time to transmit an LSA to a neighbor.

Before you begin

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **timers lsa-arrival** *msec*	Sets the LSA arrival time in milliseconds. The range is from 10 to 600000. The default is 1000 milliseconds.
Step 4	switch(config-router)# **timers lsa-group-pacing** *seconds*	Sets the interval in seconds for grouping LSAs. The range is from 1 to 1800. The default is 10 seconds.

	Command or Action	Purpose
Step 5	switch(config-router)# **timers throttle lsa** *start-time hold-interval max-time*	Sets the rate limit in milliseconds for generating LSAs. You can configure the following timers: • *start-time*—The range is from 50 to 5000 milliseconds. The default value is 50 milliseconds. • *hold-interval*—The range is from 50 to 30,000 milliseconds. The default value is 5000 milliseconds. • *max-time*—The range is from 50 to 30,000 milliseconds. The default value is 5000 milliseconds.
Step 6	switch(config-router)# **address-family ipv6 unicast**	Enters IPv6 unicast address family mode.
Step 7	switch(config-router)# **timers throttle spf** *delay-time hold-time*	Sets the SPF best path schedule initial delay time and the minimum hold time in seconds between SPF best path calculations. The range is from 1 to 600000. The default is no delay time and 5000 millisecond hold time.
Step 8	switch(config)# **interface** *type slot/port*	Enters interface configuration mode.
Step 9	switch(config-if)# **ospfv3 retransmit-interval** *seconds*	Sets the estimated time in seconds between LSAs transmitted from this interface. The range is from 1 to 65535. The default is 5.
Step 10	switch(config-if)# **ospfv3 transmit-delay** *seconds*	Sets the estimated time in seconds to transmit an LSA to a neighbor. The range is from 1 to 450. The default is 1.
Step 11	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This shows how to create a stub area that blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# timers lsa-group-pacing 300
switch(config-router)# copy running-config startup-config
```

Configuring the OSPFv3 Max-Metric Router LSA

You can configure OSPFv3 to advertise its locally generated router LSAs with the maximum metric value possible (the infinity metric 0xFFF). This feature allows OSPFv3 processes to converge but not attract transit traffic through the device if there are better alternate paths. After a specified timeout or a notification from BGP, OSPFv3 advertises the LSAs with normal metrics.

Before you begin

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **max-metric router-lsa** [**external-lsa** [*max-metric-value*]] [**stub-prefix-lsa**] [**on-startup** [*seconds*]] [**wait-for bgp** *tag*] [**inter-area-prefix-lsa** [*max-metric-value*]]	Configures a device that is running the OSPFv3 protocol to advertise a maximum metric so that other devices do not prefer the device as an intermediate hop in their SPF calculations.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to configure a router to advertise a maximum metric for the stub links:

```
switch(config)# router ospfv3 200
switch(config-router)# max-metric router-lsa stub-prefix-lsa
```

Configuring Graceful Restart

Graceful restart is enabled by default. You can configure the following optional parameters for graceful restart in an OSPFv3 instance:

- Grace period—Configures how long neighbors should wait after a graceful restart has started before tearing down adjacencies.
- Helper mode disabled—Disables helper mode on the local OSPFv3 instance. OSPFv3 does not participate in the graceful restart of a neighbor.
- Planned graceful restart only—Configures OSPFv3 to support graceful restart only in the event of a planned restart.

Before you begin

You must enable OSPF.

Ensure that all neighbors are configured for graceful restart with matching optional parameters set.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 3	switch(config-router)# **graceful restart**	Enables graceful restart. A graceful restart is enabled by default.
Step 4	switch(config-router)# **graceful-restart grace-period** *seconds*	Sets the grace period, in seconds. The range is from 5 to 1800. The default is 60 seconds.
Step 5	switch(config-router)# **graceful-restart helper-disable**	Disables helper mode. Enabled by default.
Step 6	switch(config-router)# **graceful-restart planned-only**	Configures graceful restart for planned restarts only.
Step 7	(Optional) switch(config-if)# **show ipv6 ospfv3** *instance-tag*	Displays OSPFv3 information.
Step 8	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This shows how to create a stub area that blocks all summary route updates:

```
switch# configure terminal
switch(config)# router ospfv3 201
switch(config-router)# graceful-restart
switch(config-router)# graceful-restart grace-period 120
switch(config-router)# copy running-config startup-config
```

Restarting an OSPFv3 Instance

You can restart an OSPv3 instance. This action clears all neighbors for the instance.

To restart an OSPFv3 instance and remove all associated neighbors, use the following command:

Procedure

	Command or Action	Purpose
Step 1	switch(config)# **restart ospfv3** *instance-tag*	Restarts the OSPFv3 instance and removes all neighbors.

Configuring OSPFv3 with Virtualization

You can configure multiple OSPFv3 instances in each VDC. You can also create multiple VRFs within each VDC and use the same or multiple OSPFv3 instances in each VRF. You assign an OSPFv3 interface to a VRF.

Note Configure all other parameters for an interface after you configure the VRF for an interface. Configuring a VRF for an interface deletes all the configuration for that interface.

Before you begin

Create the VDCs.

You must enable OSPF.

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **vrf context** *vrf-name*	Creates a new VRF and enters VRF configuration mode.
Step 3	switch(config)# **router ospfv3** *instance-tag*	Creates a new OSPFv3 instance with the configured instance tag.
Step 4	switch(config-router)# **vrf** *vrf-name*	Enters VRF configuration mode.
Step 5	(Optional) switch(config-router-vrf)# **maximum-paths** *paths*	Configures the maximum number of equal OSPFv3 paths to a destination in the route table for this VRF. Use this command for load balancing.
Step 6	switch(config)# **interface** *type slot/port*	Enters interface configuration mode.
Step 7	switch(config-if)# **vrf member** *vrf-name*	Adds this interface to a VRF.
Step 8	switch(config-if)# **ipv6 address** *ipv6-prefix/length*	Configures an IP address for this interface. You must do this step after you assign this interface to a VRF.
Step 9	switch(config-if)# **ipv6 ospfv3** *instance-tag* **area** *area-id*	Assigns this interface to the OSPFv3 instance and area configured.
Step 10	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

This example shows how to create a VRF and add an interface to the VRF:

```
switch# configure terminal
switch(config)# vrf context NewVRF
switch(config-vrf)# exit
switch(config)# router ospfv3 201
switch(config-router)# exit
switch(config)# interface ethernet 1/2
switch(config-if)# vrf member NewVRF
switch(config-if)# ipv6 address 2001:0DB8::1/48
switch(config-if)# ipv6 ospfv3 201 area 0
switch(config-if)# copy running-config startup-config
```

Configuring OSPFv3 Authentication at Router Level

You can enable authentication of OSPFv3 packets on a per-interface basis at the Router level using the following commands.

Before you begin

Ensure you have enabled OSPF.

Ensure that you are in the correct VDC(or use the **switchto vdc** command)

Enable the authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enable the authentication package:

switch(config)# **feature imp**

Step 3 Create a new OSPFv3 instance with the configured instance tag:

switch(config)# **router ospfv3** *instance-tag*

Step 4 Enable IPSec AH Authentication:

switch(config-router)# **authentication ipsec spi** *spi auth* [**0** | **3** | **7**] *key*

You can specify the security policy index through *spi* and define the authentication algorithm through *auth* which can be md5 or sha1. Numbers 0, 3 and 7 specify the format of *key*.

Step 5 (Optional) Display OSPFv3 information:

switch(config)# **show running-config ospfv3**

Configuring OSPFv3 Authentication at Area Level

Authentication of OSPFv3 packets is enabled on a per-interface basis at the Area level using the following commands.

Before you begin

Ensure you have enabled OSPF.

Ensure that you are in the correct VDC(or use the **switchto vdc** command)

Enable the authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enable the authentication package:

switch(config)# **feature imp**

Step 3 Create a new OSPFv3 instance with the configured instance tag:

switch(config)#**router ospfv3** *instance-tag*

Step 4 Enable IPSec AH Authentication:

switch(config-router)#**area** *area-num* **authentication ipsec spi** *spi auth* [**0** | **3** | **7**] *key*

You can specify the security policy index through *spi* and define the authentication algorithm through *auth* which can be md5 or sha1. Numbers 0, 3 and 7 specify the format of *key*.

Step 5 (Optional) Display OSPFv3 information:

switch(config)# **show running-config ospfv3**

Configuring OSPFv3 Authentication at Interface Level

You can configure the authentication of OSPFv3 packets per interface using the following commands.

Before you begin

Ensure you have enabled OSPF.

Ensure that you are in the correct VDC(or use the **switchto vdc** command)

Enable the authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enables the authentication mode:

switch(config)# **feature imp**

Step 3 Enters the interface configuration mode:

switch(config)# **interface ethernet** *interface*

Step 4 Change the port mode to Layer 3 interface:

switch(config-if)# **no switchport**

Step 5 Specify the OSPFv3 instance and area for the interface:

switch(config-if)# **ipv6 router ospfv3** *instance-tag* **area** *area-id*

Step 6 Enable IPSec AH Authentication:

switch(config-if)# **ospfv3 authentication ipsec spi** *spi auth* [**0** | **3** | **7**] *key*

You can specify the security policy index through *spi* and define the authentication algorithm through *auth* which can be md5 or sha1. Numbers 0, 3 and 7 specify the format of *key*.

Step 7 (Optional) Display the running configuration on the interface:

switch(config-if)#**show run interface** *interface*

Configuration Example

The following example shows how to enable security for Ethernet interface 2/1.

```
switch# configure terminal
switch(config)# interface ethernet 2/1
switch(config-if)# no switchport
switch(config-if)# ipv6 router ospfv3 1 area 0
switch(config-if)# ospfv3 authentication ipsec spi 256 md5 0 11111111111111111111111111111111
switch(config-if)# show run interface ethernet 2/1

!Command: show running-config interface Ethernet2/1
!Time: Mon Oct 26 09:19:30 2015

version 7.2(0)D1(1)

interface Ethernet2/1
  shutdown
  no switchport
  medium p2p
  ospfv3 authentication ipsec spi 256 md5 3 b54dc5a961fb42098f6902e512cb6e099d44
d3239f4e48e73668de6f52254f0e
  ipv6 router ospfv3 1 area 0.0.0.0

switch(config-if)#
```

Configuring OSPFv3 Encryption at Router Level

You can configure OSPFv3 ESP to encrypt and authenticate OSPFv3 packets at the router level using the following commands.

Before you begin

You must enable OSPFv3.

Enable authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enter the configuration of OSPFv3 mode:

switch# **configure ospfv3**

Step 3 Enable authentication package:

switch(config)# **feature imp**

Step 4 Create a new OSPFv3 instance with the configured instance tag:

switch(config)# **router ospfv3** *instance-tag*

Step 5 Enable IPSec ESP encryption:

switch(config-router)# **encryption ipsec spi** *spi_id* **esp** *encrypt_algorithm* [**0** | **3** | **7**] *key* **authentication** *auth_algorithm* [**0** | **3** | **7**] *key*.

You can specify the security policy index through *spi_id* and define the encryption algorithm through *encrypt_algorithm* which can be 3des, aes 128 or null. Numbers 0, 3, and 7 specify the format of the *key*. You can define the authentication algorithm through *auth_algorithm* which can be sha1 or md5.

Note MD5 is not supported in FIPS mode.

Step 6 (Optional) Display OSPFv3 information:

switch(config)# **show running-config ospfv3**

Configuring OSPFv3 Encryption at Area Level

You can configure OSPFv3 ESP to encrypt and authenticate OSPFv3 packets at the area level using the following commands.

Before you begin

You must enable OSPFv3.

Enable authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enter the configuration of OSPFv3 mode:

switch# **configure ospfv3**

Step 3 Enable the authentication package:

switch(config)# **feature imp**

Step 4 Create a new OSPFv3 instance with the configured instance tag:

switch(config)# **router ospfv3** *instance-tag*

Step 5 Enable IPSec ESP Encryption:

switch(config-router)#**area** *area-num* **encryption ipsec spi** *spi_val* **esp** *encrypt_algorithm* [**0** | **3** | **7**] *key* **authentication** *auth_algorithm* [**0** | **3** | **7**] *key*

You can specify the security policy index through *spi_id* and define the encryption algorithm through *encrypt_algorithm* which can be 3des, aes 128 or null. Numbers 0, 3, and 7 specify the format of the *key*. You can define the authentication algorithm through *auth_algorithm* which can be sha1 or md5.

Note MD5 is not supported in FIPS mode.

Step 6 (Optional) Display OSPFv3 information:

switch(config)# **show running-config ospfv3**

Configuring OSPFv3 Encryption at Interface Level

You can configure OSPFv3 ESP to encrypt and authenticate OSPFv3 packets at the interface level using the following commands.

Before you begin

You must enable OSPFv3.

Enable authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enter the configuration of OSPFv3 mode:

switch# **configure ospfv3**

Step 3 Enables the authentication mode:

switch(config)# **feature imp**

Step 4 Enters the interface configuration mode:

switch(config)# **interface ethernet** *interface*

Step 5 Specify the OSPFv3 instance and area for the interface:

switch(config-if)# **ipv6 router ospfv3** *instance-tag* **area** *area-id*

Step 6 Enable IPSec ESP Encryption:

switch(config-if)# **ospfv3 encryption ipsec spi** *spi_id* **esp** *encrypt_algorithm* [**0** | **3** | **7**] *key* **authentication** *auth_algorithm* [**0** | **3** | **7**] *key*

You can specify the security policy index through *spi_id* and define the encryption algorithm through *encrypt_algorithm* which can be 3des, aes 128 or null. Numbers 0,3 and 7 specify the format of the *key*. You can define the authentication algorithm through *auth_algorithm* which can be sha1 or md5.

Note MD5 is not supported in FIPS mode.

Step 7 (Optional) Display the running configuration on the interface:

switch(config-if)#**show run interface** *interface*

Configuration Example

The following example shows how to enable security for Ethernet interface 3/2.

```
switch# configure terminal
switch(config)# feature ospfv3
switch(config)# feature imp
switch(config)# interface ethernet 3/2
switch(config-if)# ipv6 router ospfv3 1 area 0.0.0.0
switch(config-if)# ospfv3 encryption ipsec spi 444
  esp  Specify encryption parameters
switch(config-if)# ospfv3 encryption ipsec spi 444 esp
  3des  Use the triple DES algorithim
  aes   Use the AES algorithim
  null  Use NULL authentication
switch(config-if)# ospfv3 encryption ipsec spi 444 esp aes
  128  Use the 128-bit AES algorithim
switch(config-if)# ospfv3 encryption ipsec spi 444 esp aes 128
  0     Specifies an UNENCRYPTED encryption key will follow
  3     Specifies an 3DES ENCRYPTED encryption key will follow
  7     Specifies a Cisco type 7  ENCRYPTED encryption key will follow
  WORD  The UNENCRYPTED (cleartext) encryption key
switch(config-if)# ospfv3 encryption ipsec spi 444 esp aes 128
12345678123456781234567812345678 authentication null
switch(config-if)# sh ospfv3 interface
 Ethernet3/2 is up, line protocol is up
    IPv6 address 1:1:1:1::2/64
    Process ID 1 VRF default, Instance ID 0, area 0.0.0.0
    Enabled by interface configuration
    State DOWN, Network type BROADCAST, cost 40
    ESP Encryption AES, Authentication NULL, SPI 444, ConnId 444
switch(config-if)#
```

Configuring OSPFv3 Encryption for Virtual Links

You can configure OSPFv3 ESP to encrypt and authenticate OSPFv3 packets for virtual links using the following commands.

Before you begin

You must enable OSPFv3.

Enable authentication package.

Procedure

Step 1 Enter the global configuration mode:

switch# **configure terminal**

Step 2 Enter the configuration of OSPFv3 mode:

switch# **configure ospfv3**

Step 3 Enable the authentication package:

switch(config)# **feature imp**

Step 4 Create a new OSPFv3 instance with the configured instance tag:

switch(config)#**router ospfv3** *instance-tag*

Step 5 Enable IPSec ESP Encryption:

switch(config-router)# **encryption ipsec spi** *spi_id* **esp** *encrypt_algorithm* [**0** | **3** | **7**] *key* **authentication** *auth_algorithm* [**0** | **3** | **7**] *key*

You can specify the security policy index through *spi_id* and define the encryption algorithm through *encrypt_algorithm* which can be 3des, aes 128 or null. Numbers 0,3 and 7 specify the format of the *key*. You can define the authentication algorithm through *auth_algorithm* which can be sha1 or md5.

Note MD5 is not supported in FIPS mode.

Step 6 (Optional) Display OSPFv3 information:

switch(config)# **show running-config ospfv3**

Configuration Example

The following example shows how to encrypt Virtual links.

```
switch(config)# feature ospfv3
switch(config)# feature imp
switch(config-if)# router ospfv3 1
switch(config-router)# area 0.0.0.1 virtual-link 3.3.3.3
switch(config-router-vlink)# encryption ipsec spi ?
<256-4294967295> SPI Value
switch(config-router-vlink)# encryption ipsec spi 256 esp ?
3des Use the triple DES algorithim
aes Use the AES algorithim
null Use NULL authentication
switch(config-router-vlink)# encryption ipsec spi 256 esp aes 128
123456789A123456789B123456789C12 authentication ?
null Use NULL authentication
sha1 Use the SHA1 algorithim
switch(config-router-vlink)# encryption ipsec spi 256 esp aes 128
123456789A123456789B123456789C12 authentication null
```

Configuration Examples for OSPFv3

This example shows how to configure OSPFv3:

```
This example shows how to configure OSPFv3:
feature ospfv3
router ospfv3 201
 router-id 290.0.2.1

interface ethernet 1/2
 ipv6 address 2001:0DB8::1/48
 ipv6 ospfv3 201 area 0.0.0.10
```

Related Documents for OSPFv3

Related Topic	Document Title
OSPFv3 CLI commands	*Cisco Nexus 7000 Series NX-OS Unicast Routing Command Reference*
VDCs	*Cisco Nexus 7000 Series NX-OS Virtual Device Context Command Reference*

Feature History for OSPFv3

This table includes only the updates for those releases that have resulted in additions or changes to the feature.

Table 2: Feature History for OSPFv3

Feature Name	Release	Feature Information
OSPFv3 ESP Encryption	8.4(4)	Added ESP encryption for OSPFv3 packets.
OSPF—Distribute List to Filter Paths	6.2(6a)	Added support for filtering next-hop paths for an OSPF route to prevent the path from being added to the RIB.
Administrative distance of routes	6.2(2)	Added the **filter** keyword to the **table-map** command to specify that only routes permitted by the route map are downloaded to the RIB.
Route summarization	6.2(2)	Added the ability to prevent discard routes from being created.
OSPFv3	6.2(2)	• Bidirectional Forwarding Detection (BFD) was enhanced to add a client for OSPFv3 • Added the ability to advertise locally generated router LSAs with the maximum metric value possible. • Added the optional **name-lookup** parameter for OSPFv3 instances.
MIBs	6.2(2)	Added OSPFv3 SNMP/trap support.
OSPFv3	6.1(1)	Added support for configuring the administrative distance of routes for OSPFv3.
Passive interface	5.2(1)	Added support for setting the passive interface mode on all interfaces in the router or VRF.
OSPFv3	4.0(1)	This feature was introduced.