Implementing OSPF for IPv6

First Published: March 17, 2003
Last Updated: May 1, 2006

The *Implementing OSPF for IPv6* module expands on OSPF to provide support for IPv6 routing prefixes. This module describes the concepts and tasks you need to implement OSPF for IPv6 on your network.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all of the features documented in this module. To reach links to specific feature documentation in this module and to see a list of the releases in which each feature is supported, use the "Feature Information for Implementing OSPF for IPv6" section on page 209 or the "Start Here: Cisco IOS Software Release Specifics for IPv6 Features" document.

Finding Support Information for Platforms and Cisco IOS and Catalyst OS Software Images

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

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for Implementing OSPF for IPv6

Before you enable OSPF for IPv6 on an interface, you must do the following:

- Complete the OSPF network strategy and planning for your IPv6 network. For example, you must decide whether multiple areas are required.
- Enable IPv6 unicast routing.
- Enable IPv6 on the interface.
- Configure the IP Security (IPSec) secure socket application program interface (API) on OSPF for IPv6 in order to enable authentication and encryption.

This document assumes that you are familiar with IPv4. Refer to the publications referenced in the "Related Documents" section for IPv4 configuration and command reference information.

Restrictions for Implementing OSPF for IPv6

- When running a dual-stack IP network with OSPF version 2 for IPv4 and OSPF for IPv6, be careful when changing the defaults for commands used to enable OSPF for IPv6. Changing these defaults may affect your OSPF for IPv6 network, possibly adversely.
- Authentication is supported as of Cisco IOS Release 12.3(4)T.
- ESP authentication and encryption are supported as of Cisco IOS Release 12.4(9)T.

Information About Implementing OSPF for IPv6

To implement OSPF for IPv6, you need to understand the following concepts:

How OSPF for IPv6 Works

OSPF is a routing protocol for IP. It is a link-state protocol, as opposed to a distance-vector protocol. Think of a link as being an interface on a networking device. A link-state protocol makes its routing decisions based on the states of the links that connect source and destination machines. The state of a link is a description of that interface and its relationship to its neighboring networking devices. The interface information includes the IPv6 prefix of the interface, the network mask, the type of network it is connected to, the routers connected to that network, and so on. This information is propagated in various type of LSAs.

A router's collection of LSA data is stored in a link-state database. The contents of the database, when subjected to the Dijkstra algorithm, result in the creation of the OSPF routing table. The difference between the database and the routing table is that the database contains a complete collection of raw data; the routing table contains a list of shortest paths to known destinations via specific router interface ports.

OSPF version 3, which is described in RFC 2740, supports IPv6.

Comparison of OSPF for IPv6 and OSPF Version 2

Much of the OSPF for IPv6 feature is the same as in OSPF version 2. OSPF version 3 for IPv6, which is described in RFC 2740, expands on OSPF version 2 to provide support for IPv6 routing prefixes and the larger size of IPv6 addresses.

In OSPF for IPv6, a routing process does not need to be explicitly created. Enabling OSPF for IPv6 on an interface will cause a routing process, and its associated configuration, to be created.

In OSPF for IPv6, each interface must be enabled using commands in interface configuration mode. This feature is different from OSPF version 2, in which interfaces are indirectly enabled using the router configuration mode.

When using a nonbroadcast multiaccess (NBMA) interface in OSPF for IPv6, users must manually configure the router with the list of neighbors. Neighboring routers are identified by their router ID.

In IPv6, users can configure many address prefixes on an interface. In OSPF for IPv6, all address prefixes on an interface are included by default. Users cannot select some address prefixes to be imported into OSPF for IPv6; either all address prefixes on an interface are imported, or no address prefixes on an interface are imported.

Unlike OSPF version 2, multiple instances of OSPF for IPv6 can be run on a link.

In OSPF for IPv6, it is possible that no IPv4 addresses will be configured on any interface. In this case, the user must use the **router-id** command to configure a router ID before the OSPF process will be started. A router ID is a 32-bit opaque number. OSPF version 2 takes advantage of the 32-bit IPv4 address to pick an IPv4 address as the router ID. If an IPv4 address does exist when OSPF for IPv6 is enabled on an interface, then that IPv4 address is used for the router ID. If more than one IPv4 address is available, a router ID is chosen using the same rules as for OSPF version 2.

OSPF automatically prefers a loopback interface over any other kind, and it chooses the highest IP address among all loopback interfaces. If no loopback interfaces are present, the highest IP address in the router is chosen. You cannot tell OSPF to use any particular interface.

For further information about configuring a router ID and the **router-id** command, refer to "Configuring OSPF" chapter of the *Cisco IOS IP Configuration Guide* and *Cisco IOS IP Command Reference, Volume 2 of 4: Routing Protocols*, Release 12.4.

LSA Types for IPv6

The following list describes LSA types, each of which has a different purpose:

- Router LSAs (Type 1)—Describes the link state and costs of a router's links to the area. These LSAs are flooded within an area only. The LSA indicates if the router is an Area Border Router (ABR) or Autonomous System Boundary Router (ASBR), and if it is one end of a virtual link. Type 1 LSAs are also used to advertise stub networks. In OSPF for IPv6, these LSAs have no address information and are network-protocol-independent. In OSPF for IPv6, router interface information may be spread across multiple router LSAs. Receivers must concatenate all router LSAs originated by a given router when running the SPF calculation.

- Network LSAs (Type 2)—Describes the link-state and cost information for all routers attached to the network. This LSA is an aggregation of all the link-state and cost information in the network. Only a designated router tracks this information and can generate a network LSA. In OSPF for IPv6, network LSAs have no address information and are network-protocol-independent.
- Interarea-prefix LSAs for ABRs (Type 3)—Advertises internal networks to routers in other areas (interarea routes). Type 3 LSAs may represent a single network or a set of networks summarized into one advertisement. Only ABRs generate summary LSAs. In OSPF for IPv6, addresses for these LSAs are expressed as *prefix*, *prefix length* instead of *address*, *mask*. The default route is expressed as a prefix with length 0.
- Interarea-router LSAs for ASBRs (Type 4)—Advertise the location of an ASBR. Routers that are trying to reach an external network use these advertisements to determine the best path to the next hop. ASBRs generate Type 4 LSAs.
- Autonomous system external LSAs (Type 5)—Redistributes routes from another AS, usually from a different routing protocol into OSPF. In OSPF for IPv6, addresses for these LSAs are expressed as *prefix*, *prefix length* instead of *address*, *mask*. The default route is expressed as a prefix with length 0.
- Link LSAs (Type 8)—Have local-link flooding scope and are never flooded beyond the link with which they are associated. Link LSAs provide the link-local address of the router to all other routers attached to the link, inform other routers attached to the link of a list of IPv6 prefixes to associate with the link, and allow the router to assert a collection of Options bits to associate with the network LSA that will be originated for the link.
- Intra-Area-Prefix LSAs (Type 9)—A router can originate multiple intra-area-prefix LSAs for each router or transit network, each with a unique link-state ID. The link-state ID for each intra-area-prefix LSA describes its association to either the router LSA or the network LSA and contains prefixes for stub and transit networks.

An address prefix occurs in almost all newly defined LSAs. The prefix is represented by three fields: PrefixLength, PrefixOptions, and Address Prefix. In OSPF for IPv6, addresses for these LSAs are expressed as *prefix*, *prefix length* instead of *address*, *mask*. The default route is expressed as a prefix with length 0. Type 3 and Type 9 LSAs carry all IPv6 prefix information that, in IPv4, is included in router LSAs and network LSAs. The Options field in certain LSAs (router LSAs, network LSAs, interarea-router LSAs, and link LSAs) has been expanded to 24 bits to provide support for OSPF in IPv6.

In OSPF for IPv6, the sole function of link-state ID in interarea-prefix LSAs, interarea-router LSAs, and autonomous-system external LSAs is to identify individual pieces of the link-state database. All addresses or router IDs that are expressed by the link-state ID in OSPF version 2 are carried in the body of the LSA in OSPF for IPv6.

The link-state ID in network LSAs and link LSAs is always the interface ID of the originating router on the link being described. For this reason, network LSAs and link LSAs are now the only LSAs whose size cannot be limited. A network LSA must list all routers connected to the link, and a link LSA must list all of the address prefixes of a router on the link.

NBMA in OSPF for IPv6

On NBMA networks, the designated router (DR) or backup DR (BDR) performs the LSA flooding. On point-to-point networks, flooding simply goes out an interface directly to a neighbor.

Routers that share a common segment (Layer 2 link between two interfaces) become neighbors on that segment. OSPF uses the Hello protocol, periodically sending hello packets out each interface. Routers become neighbors when they see themselves listed in the neighbor's hello packet. After two routers become neighbors, they may proceed to exchange and synchronize their databases, which creates an adjacency. Not all neighboring routers have an adjacency.

On point-to-point and point-to-multipoint networks, the software floods routing updates to immediate neighbors. There is no DR or BDR; all routing information is flooded to each networking device.

On broadcast or NBMA segments only, OSPF minimizes the amount of information being exchanged on a segment by choosing one router to be a DR and one router to be a BDR. Thus, the routers on the segment have a central point of contact for information exchange. Instead of each router exchanging routing updates with every other router on the segment, each router exchanges information with the DR and BDR. The DR and BDR relay the information to the other routers.

The software looks at the priority of the routers on the segment to determine which routers will be the DR and BDR. The router with the highest priority is elected the DR. If there is a tie, then the router with the higher router ID takes precedence. After the DR is elected, the BDR is elected the same way. A router with a router priority set to zero is ineligible to become the DR or BDR.

When using NBMA in OSPF for IPv6, you cannot automatically detect neighbors. On an NBMA interface, you must configure your neighbors manually using interface configuration mode.

Force SPF in OSPF for IPv6

When the **process** keyword is used with the **clear ipv6 ospf** command, the OSPF database is cleared and repopulated, and then the SPF algorithm is performed. When the **force-spf** keyword is used with the **clear ipv6 ospf** command, the OSPF database is not cleared before the SPF algorithm is performed.

Load Balancing in OSPF for IPv6

When a router learns multiple routes to a specific network via multiple routing processes (or routing protocols), it installs the route with the lowest administrative distance in the routing table. Sometimes the router must select a route from among many learned via the same routing process with the same administrative distance. In this case, the router chooses the path with the lowest cost (or metric) to the destination. Each routing process calculates its cost differently and the costs may need to be manipulated in order to achieve load balancing.

OSPF performs load balancing automatically in the following way. If OSPF finds that it can reach a destination through more than one interface and each path has the same cost, it installs each path in the routing table. The only restriction on the number of paths to the same destination is controlled by the **maximum-paths** command. The default maximum paths is 16, and the range is from 1 to 64.

Importing Addresses into OSPF for IPv6

When importing the set of addresses specified on an interface on which OSPF for IPv6 is running into OSPF for IPv6, users cannot select specific addresses to be imported. Either all addresses are imported, or no addresses are imported.

OSPF for IPv6 Customization

You can customize OSPF for IPv6 for your network, but you likely will not need to do so. The defaults for OSPF in IPv6 are set to meet the requirements of most customers and features. If you must change the defaults, refer to the IPv4 configuration guide and the IPv6 command reference to find the appropriate syntax.

Caution Be careful when changing the defaults. Changing defaults will affect your OSPF for IPv6 network, possibly adversely.

OSPF for IPv6 Authentication Support with IPSec

In order to ensure that OSPF for IPv6 packets are not altered and re-sent to the router, causing the router to behave in a way not desired by its managers, OSPF for IPv6 packets must be authenticated. OSPF for IPv6 uses the IP Security (IPSec) secure socket application program interface (API) to add authentication to OSPF for IPv6 packets. This API has been extended to provide support for IPv6.

OSPF for IPv6 requires the use of IPSec to enable authentication. Crypto images are required to use authentication, because only crypto images include the IPSec API needed for use with OSPF for IPv6.

In OSPF for IPv6, authentication fields have been removed from OSPF headers. When OSPF runs on IPv6, OSPF requires the IPv6 authentication header (AH) or IPv6 ESP header to ensure integrity, authentication, and confidentiality of routing exchanges. IPv6 AH and ESP extension headers can be used to provide authentication and confidentiality to OSPF for IPv6.

To use the IPSec AH, you must enable the **ipv6 ospf authentication** command. To use the IPSec ESP, you must enable the **ipv6 ospf encryption** command. The ESP header may be applied alone or in combination with the AH, and when ESP is used, both encryption and authentication are provided. Security services can be provided between a pair of communicating hosts, between a pair of communicating security gateways, or between a security gateway and a host.

To configure IPSec, users configure a security policy, which is a combination of the security policy index (SPI) and the key (the key is used to create and validate the hash value). IPSec for OSPF for IPv6 can be configured on an interface or on an OSPF area. For higher security, users should configure a different policy on each interface configured with IPSec. If a user configures IPSec for an OSPF area, the policy is applied to all of the interfaces in that area, except for the interfaces that have IPSec configured directly. Once IPSec is configured for OSPF for IPv6, IPSec is invisible to the user.

The secure socket API is used by applications to secure traffic. The API needs to allow the application to open, listen, and close secure sockets. The binding between the application and the secure socket layer also allows the secure socket layer to inform the application of changes to the socket, such as connection open and close events. The secure socket API is able to identify the socket; that is, it can identify the local and remote addresses, masks, ports, and protocol that carry the traffic requiring security.

Each interface has a secure socket state, which can be one of the following:

- NULL: Do not create a secure socket for the interface if authentication is configured for the area.
- DOWN: IPSec has been configured for the interface (or the area that contains the interface), but OSPF for IPv6 either has not requested IPSec to create a secure socket for this interface, or there is an error condition.
- GOING UP: OSPF for IPv6 has requested a secure socket from IPSec and is waiting for a CRYPTO_SS_SOCKET_UP message from IPSec.
- UP: OSPF has received a CRYPTO_SS_SOCKET_UP message from IPSec.

- CLOSING: The secure socket for the interface has been closed. A new socket may be opened for the interface, in which case the current secure socket makes the transition to the DOWN state. Otherwise, the interface will become UNCONFIGURED.
- UNCONFIGURED: Authentication is not configured on the interface.

OSPF will not send or accept packets while in the DOWN state.

For further information on IPSec, refer to the *Implementing IPSec in IPv6 Security* document.

OSPF for IPv6 Virtual Links

For each virtual link, a master security information datablock is created for the virtual link. Because a secure socket must be opened on each interface, there will be a corresponding security information datablock for each interface in the transit area. The secure socket state is kept in the interface's security information datablock. The state field in the master security information datablock reflects the status of all of the secure sockets opened for the virtual link. If all of the secure sockets are UP, then the security state for the virtual link will be set to UP.

Packets sent on a virtual link with IPSec must use predetermined source and destination addresses. The first local area address found in the router's intra-area-prefix LSA for the area is used as the source address. This source address is saved in the area data structure and used when secure sockets are opened and packets sent over the virtual link. The virtual link will not transition to the point-to-point state until a source address is selected. Also, when the source or destination address changes, the previous secure sockets must be closed and new secure sockets opened.

For further information on IPSec and how to implement it, refer to the *Implementing Security for IPv6* module.

How to Implement OSPF for IPv6

This section contains the following procedures:

- Enabling OSPF for IPv6 on an Interface, page 193 (required)
- Defining an OSPF for IPv6 Area Range, page 194 (optional)
- Configuring IPSec on OSPF for IPv6, page 195 (optional)
- Configuring NBMA Interfaces, page 200 (optional)
- Forcing an SPF Calculation, page 201 (optional)
- Verifying OSPF for IPv6 Configuration and Operation, page 202 (optional)

Enabling OSPF for IPv6 on an Interface

This task explains how to enable OSPF for IPv6 routing and configure OSPF for IPv6 on each interface. By default, OSPF for IPv6 routing is disabled and OSPF for IPv6 is not configured on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface ethernet 0/0`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 4	**ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*] **Example:** `Router(config-if)# ipv6 ospf 1 area 0`	Enables OSPF for IPv6 on an interface.

Defining an OSPF for IPv6 Area Range

The cost of the summarized routes will be the highest cost of the routes being summarized. For example, if the following routes are summarized:

```
OI  2001:0DB8:0:0:7::/64 [110/20]
     via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0
OI  2001:0DB8:0:0:8::/64 [110/100]
     via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0
OI  2001:0DB8:0:0:9::/64 [110/20]
     via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0
```

They becomes one summarized route, as follows:

```
OI  2001:0DB8::/48 [110/100]
     via FE80::A8BB:CCFF:FE00:6F00, Ethernet0/0
```

This task explains how to consolidate or summarize routes for an OSPF area.

Prerequisites

OSPF for IPv6 routing must be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **ipv6 router ospf** *process-id*
4. **area** *area-id* **range** *ipv6-prefix/prefix-length* [**advertise** | **not-advertise**] [**cost** *cost*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router ospf** *process-id* **Example:** `Router(config)# ipv6 router ospf 1`	Enables OSPF router configuration mode.
Step 4	**area** *area-id* **range** *ipv6-prefix/prefix-length* [**advertise** | **not-advertise**] [**cost** *cost*] **Example:** `Router(config-rtr)# area range 1 2001:0DB8::/48`	Consolidates and summarizes routes at an area boundary.

Configuring IPSec on OSPF for IPv6

Once you have configured OSPF for IPv6 and decided on your authentication, you must define the security policy on each of the routers within the group. The security policy consists of the combination of the key and the SPI. To define a security policy, you must define an SPI and a key.

You can configure an authentication or encryption policy either on an interface or for an OSPF area. When you configure for an area, the security policy is applied to all of the interfaces in the area. For higher security, use a different policy on each interface.

You can configure authentication and encryption on virtual links.

The following tasks explain how to configure authentication and encryption on an interface or in an OSPF area, and on virtual links.

- Defining Authentication on an Interface, page 195
- Defining Encryption on an Interface, page 196
- Defining Authentication in an OSPF Area, page 197
- Defining Encryption in an OSPF Area, page 198
- Defining Authentication and Encryption for a Virtual Link in an OSPF Area, page 199

Defining Authentication on an Interface

This task explains how to define authentication on an interface.

Prerequisites

Before you configure IPSec on an interface, you must configure OSPF for IPv6 on that interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ipv6 ospf authentication ipsec spi** *spi* **md5** [*key-encryption-type* {*key* | **null**}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface ethernet 0/0`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 4	**ipv6 ospf authentication ipsec spi** *spi* **md5** [*key-encryption-type* {*key* | **null**}] **Example:** `Router(config-if)# ipv6 ospf authentication ipsec spi 500 md5 1234567890abcdef1234567890abcdef`	Specifies the authentication type for an interface.

Defining Encryption on an Interface

This task describes how to define encryption on an interface.

Prerequisites

Before you configure IPSec on an interface, you must configure OSPF for IPv6 on that interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **ipv6 ospf encryption** {**ipsec spi** *spi* **esp** *encryption-algorithm* [[*key-encryption-type*] *key*] *authentication-algorithm* [*key-encryption-type*] *key* | **null**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**`enable`** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**`configure terminal`** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**`interface`** *`type number`* **Example:** `Router(config)# interface ethernet 0/0`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 4	**`ipv6 ospf encryption`** {**`ipsec spi`** *`spi`* **`esp`** *`encryption-algorithm`* [[*`key-encryption-type`*] *`key`*] *`authentication-algorithm`* [*`key-encryption-type`*] *`key`* | **`null`**} **Example:** `Router(config-if) ipv6 ospf encryption ipsec spi 1001 esp null sha1 123456789A123456789B123456789C123456789D`	Specifies the encryption type for an interface.

Defining Authentication in an OSPF Area

This task explains how to define authentication in an OSPF area.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf** *process-id*
4. **area** *area-id* **authentication ipsec spi** *spi* **md5** [*key-encryption-type*] *key*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router ospf** *process-id* **Example:** `Router(config)# ipv6 router ospf 1`	Enables OSPF router configuration mode.
Step 4	**area** *area-id* **authentication ipsec spi** *spi* **md5** [*key-encryption-type*] *key* **Example:** `Router(config-rtr)# area 1 authentication ipsec spi 678 md5 1234567890ABCDEF1234567890ABCDEF`	Enables authentication in an OSPF area.

Defining Encryption in an OSPF Area

This task describes how to define encryption in an OSPF area.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf** *process-id*
4. **area** *area-id* **encryption ipsec spi** *spi* **esp** *encryption-algorithm* [[*key-encryption-type*] *key*] *authentication-algorithm* [*key-encryption-type*] *key*

DETAILED STEPS

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

	Command or Action	Purpose
Step 3	**ipv6 router ospf** *process-id* **Example:** `Router(config)# ipv6 router ospf 1`	Enables OSPF router configuration mode.
Step 4	**area** *area-id* **encryption ipsec spi** *spi* **esp** *encryption-algorithm* [[*key-encryption-type*] *key*] *authentication-algorithm* [*key-encryption-type*] *key* **Example:** `Router(config-rtr)# area 1 encryption ipsec spi 500 esp null md5 1aaa2bbb3ccc4ddd5eee6fff7aaa8bbb`	Enables encryption in an OSPF area.

Defining Authentication and Encryption for a Virtual Link in an OSPF Area

The following task describes how to define authentication and encryption for virtual links in an OSPF area.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 router ospf** *process-id*
4. **area** *area-id* **virtual-link** *router-id* **authentication ipsec spi** *spi authentication-algorithm* [*key-encryption-type*] *key*
5. **area** *area-id* **virtual-link** *router-id* **encryption ipsec spi** *spi* **esp** *encryption-algorithm* [[*key-encryption-type*] *key*] *authentication-algorithm* [*key-encryption-type*] *key*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 router ospf** *process-id* **Example:** `Router(config)# ipv6 router ospf 1`	Enables OSPF router configuration mode.

	Command or Action	Purpose
Step 4	**area** *area-id* **virtual-link** *router-id* **authentication ipsec spi** *spi* *authentication-algorithm* [*key-encryption-type*] *key* **Example:** `Router(config-rtr)# area 1 virtual-link 10.0.0.1 authentication ipsec spi 940 md5 1234567890ABCDEF1234567890ABCDEF`	Enables authentication for virtual links in an OSPF area.
Step 5	**area** *area-id* **virtual-link** *router-id* **encryption ipsec spi** *spi* **esp** *encryption-algorithm* [[*key-encryption-type*] *key*] *authentication-algorithm* [*key-encryption-type*] *key* **Example:** `Router(config-rtr)# area 1 virtual-link 10.1.0.1 hello-interval 2 dead-interval 10 encryption ipsec spi 3944 esp null sha1 123456789A123456789B123456789C123456789D`	Enables encryption for virtual links in an OSPF area.

Configuring NBMA Interfaces

You can customize OSPF for IPv6 in your network to use NBMA interfaces. OSPF for IPv6 cannot automatically detect neighbors over NBMA interfaces. On an NBMA interface, you must configure your neighbors manually using interface configuration mode. This task explains how to configure NBMA interfaces.

Prerequisites

Before you configure NBMA interfaces, you must perform the following tasks:

- Configure your network to be an NBMA network
- Identify each neighbor

Restrictions

- You cannot automatically detect neighbors when using NBMA interfaces. You must manually configure your router to detect neighbors when using an NBMA interface.
- When configuring the **ipv6 ospf neighbor** command, the IPv6 address used must be the link-local address of the neighbor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*

4. **frame-relay map ipv6** *ipv6-address dlci* [**broadcast**] [**cisco**] [**ietf**] [**payload-compression** {**packet-by-packet** | **frf9 stac** [*hardware-options*] | **data-stream stac** [*hardware-options*]}]
5. **ipv6 ospf neighbor** *ipv6-address* [**priority** *number*] [**poll-interval** *seconds*] [**cost** *number*] [**database-filter all out**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Router(config)# interface serial 0`	Specifies an interface type and number, and places the router in interface configuration mode.
Step 4	**frame-relay map ipv6** *ipv6-address dlci* [**broadcast**] [**cisco**] [**ietf**] [**payload-compression** {**packet-by-packet** | **frf9 stac** [*hardware-options*] | **data-stream stac** [*hardware-options*]}] **Example:** `Router(config-if)#  frame-relay map ipv6 FE80::A8BB:CCFF:FE00:C01 120`	Defines the mapping between a destination IPv6 address and the data-link connection identifier (DLCI) used to connect to the destination address. • In this example, the NBMA link is frame relay. For other kinds of NBMA links, different mapping commands are used.
Step 5	**ipv6 ospf neighbor** *ipv6-address* [**priority** *number*] [**poll-interval** *seconds*] [**cost** *number*] [**database-filter all out**] **Example:** `Router(config-if) ipv6 ospf neighbor FE80::A8BB:CCFF:FE00:C01`	Configures an OSPF for IPv6 neighboring router.

Forcing an SPF Calculation

This task explains how to start the SPF algorithm without first clearing the OSPF database.

SUMMARY STEPS

1. **enable**
2. **clear ipv6 ospf** [*process-id*] {**process** | **force-spf** | **redistribution**}

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**clear ipv6 ospf** [*process-id*] {**process** | **force-spf** | **redistribution**} **Example:** `Router# clear ipv6 ospf force-spf`	Clears the OSPF state based on the OSPF routing process ID.

Verifying OSPF for IPv6 Configuration and Operation

This task explains how to display information to verify the configuration and operation of OSPF for IPv6.

SUMMARY STEPS

1. **enable**
2. **show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-type interface-number*]
3. **show ipv6 ospf** [*process-id*] [*area-id*]
4. **show crypto ipsec policy** [**name** *policy-name*]
5. **show crypto ipsec sa** [**map** *map-name* | **address** | **identity** | **interface** *interface-type interface-number* | **peer** [**vrf** *fvrf-name*] **address** | **vrf** *ivrf-name* | **ipv6** [*interface-type interface-number*]] [**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-type interface-number*] **Example:** `Router# show ipv6 ospf interface`	Displays OSPF-related interface information.
Step 3	**show ipv6 ospf** [*process-id*] [*area-id*] **Example:** `Router# show ipv6 ospf`	Displays general information about OSPF routing processes.

	Command or Action	Purpose
Step 4	**show crypto ipsec policy** [**name** *policy-name*] **Example:** `Router# show crypto ipsec policy`	Displays the parameters for each IPSec parameter.
Step 5	**show crypto ipsec sa** [**map** *map-name* | **address** | **identity** | **interface** *interface-type interface-number* | **peer** [**vrf** *fvrf-name*] **address** | **vrf** *ivrf-name* | **ipv6** [*interface-type interface-number*]] [**detail**] **Example:** `Router# show crypto ipsec sa ipv6`	Displays the settings used by current security associations (SAs).

Examples

This section provides the following output examples:

- Sample Output for the show ipv6 ospf interface Command, page 203
- Sample Output for the show ipv6 ospf Command, page 204
- Sample Output for the show crypto ipsec policy Command, page 205
- Sample Output for the show crypto ipsec sa ipv6 Command, page 205

Sample Output for the show ipv6 ospf interface Command

The following is sample output from the **show ipv6 ospf interface** command with regular interfaces and a virtual link that are protected by encryption and authentication:

```
Router# show ipv6 ospf interface

OSPFv3_VL1 is up, line protocol is up
   Interface ID 69
   Area 0, Process ID 1, Instance ID 0, Router ID 10.0.0.1
   Network Type VIRTUAL_LINK, Cost: 64
   Configured as demand circuit.
   Run as demand circuit.
   DoNotAge LSA allowed.
   NULL encryption SHA-1 auth SPI 3944, secure socket UP (errors: 0)
   Transmit Delay is 1 sec, State POINT_TO_POINT,
   Timer intervals configured, Hello 2, Dead 10, Wait 40, Retransmit 5
     Hello due in 00:00:00
   Index 1/3/5, flood queue length 0
   Next 0x0(0)/0x0(0)/0x0(0)
   Last flood scan length is 1, maximum is 1
   Last flood scan time is 0 msec, maximum is 0 msec
   Neighbor Count is 1, Adjacent neighbor count is 1
     Adjacent with neighbor 10.2.0.1  (Hello suppressed)
   Suppress hello for 1 neighbor(s)
OSPFv3_VL0 is up, line protocol is up
   Interface ID 67
   Area 0, Process ID 1, Instance ID 0, Router ID 10.0.0.1
   Network Type VIRTUAL_LINK, Cost: 128
   Configured as demand circuit.
   Run as demand circuit.
   DoNotAge LSA allowed.
   MD5 authentication SPI 940, secure socket UP (errors: 0)
   Transmit Delay is 1 sec, State POINT_TO_POINT,
```

```
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
     Hello due in 00:00:09
   Index 1/2/4, flood queue length 0
   Next 0x0(0)/0x0(0)/0x0(0)
   Last flood scan length is 1, maximum is 10
   Last flood scan time is 0 msec, maximum is 0 msec
   Neighbor Count is 1, Adjacent neighbor count is 1
     Adjacent with neighbor 10.1.0.1  (Hello suppressed)
   Suppress hello for 1 neighbor(s)
Ethernet1/0 is up, line protocol is up
   Link Local Address FE80::A8BB:CCFF:FE00:6601, Interface ID 6
   Area 0, Process ID 1, Instance ID 0, Router ID 10.0.0.1
   Network Type BROADCAST, Cost: 10
   Transmit Delay is 1 sec, State DR, Priority 1
   Designated Router (ID) 10.0.0.1, local address FE80::A8BB:CCFF:FE00:6601
   No backup designated router on this network
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
     Hello due in 00:00:09
   Index 1/1/1, flood queue length 0
   Next 0x0(0)/0x0(0)/0x0(0)
   Last flood scan length is 0, maximum is 0
   Last flood scan time is 0 msec, maximum is 0 msec
   Neighbor Count is 0, Adjacent neighbor count is 0
   Suppress hello for 0 neighbor(s)
Serial12/0 is up, line protocol is up
   Link Local Address FE80::A8BB:CCFF:FE00:6600, Interface ID 50
   Area 1, Process ID 1, Instance ID 0, Router ID 10.0.0.1
   Network Type POINT_TO_POINT, Cost: 64
   AES-CBC encryption SHA-1 auth SPI 2503, secure socket UP (errors: 0)
   authentication NULL
   Transmit Delay is 1 sec, State POINT_TO_POINT,
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
     Hello due in 00:00:09
   Index 1/2/3, flood queue length 0
   Next 0x0(0)/0x0(0)/0x0(0)
   Last flood scan length is 1, maximum is 5
   Last flood scan time is 0 msec, maximum is 0 msec
   Neighbor Count is 1, Adjacent neighbor count is 1
     Adjacent with neighbor 10.2.0.1
   Suppress hello for 0 neighbor(s)
Serial11/0 is up, line protocol is up
   Link Local Address FE80::A8BB:CCFF:FE00:6600, Interface ID 46
   Area 1, Process ID 1, Instance ID 0, Router ID 10.0.0.1
   Network Type POINT_TO_POINT, Cost: 64
   MD5 authentication (Area) SPI 500, secure socket UP (errors: 0)
   Transmit Delay is 1 sec, State POINT_TO_POINT,
   Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
     Hello due in 00:00:09
   Index 1/1/2, flood queue length 0
   Next 0x0(0)/0x0(0)/0x0(0)
   Last flood scan length is 1, maximum is 5
   Last flood scan time is 0 msec, maximum is 0 msec
   Neighbor Count is 1, Adjacent neighbor count is 1
     Adjacent with neighbor 1.0.0.1
   Suppress hello for 0 neighbor(s)
```

Sample Output for the show ipv6 ospf Command

The following is sample output from the **show ipv6 ospf** command:

```
Router# show ipv6 ospf

Routing Process "ospfv3 1" with ID 172.16.3.3
 It is an autonomous system boundary router
```

```
 Redistributing External Routes from,
    static
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 1. Checksum Sum 0x218D
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
    Area 1
        Number of interfaces in this area is 2
        SPF algorithm executed 9 times
        Number of LSA 15. Checksum Sum 0x67581
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

Sample Output for the show crypto ipsec policy Command

The following is sample output from the **show crypto ipsec policy** command:

```
Router# show crypto ipsec policy

Crypto IPsec client security policy data

Policy name:      OSPFv3-1-1000
Policy refcount: 1
Inbound  AH SPI: 1000 (0x3E8)
Outbound AH SPI: 1000 (0x3E8)
Inbound  AH Key: 1234567890ABCDEF1234567890ABCDEF
Outbound AH Key: 1234567890ABCDEF1234567890ABCDEF
Transform set:   ah-md5-hmac
```

Sample Output for the show crypto ipsec sa ipv6 Command

The following is sample output from the **show crypto ipsec sa ipv6** command:

```
Router# show crypto ipsec sa ipv6

IPv6 IPsec SA info for interface Ethernet0/0

   protected policy name:OSPFv3-1-1000
   IPsec created ACL name:Ethernet0/0-ipsecv6-ACL

   local  ident (addr/prefixlen/proto/port):(FE80::/10/89/0)
   remote ident (addr/prefixlen/proto/port):(::/0/89/0)
   current_peer:::
     PERMIT, flags={origin_is_acl,}
    #pkts encaps:21, #pkts encrypt:0, #pkts digest:21
    #pkts decaps:20, #pkts decrypt:0, #pkts verify:20
    #pkts compressed:0, #pkts decompressed:0
    #pkts not compressed:0, #pkts compr. failed:0
    #pkts not decompressed:0, #pkts decompress failed:0
    #send errors 0, #recv errors 0

    local crypto endpt. ::, remote crypto endpt. ::
    path mtu 1500, media mtu 1500
    current outbound spi:0x3E8(1000)

     inbound ESP SAs:

     inbound AH SAs:
      spi:0x3E8(1000)
        transform:ah-md5-hmac ,
```

```
   in use settings ={Transport, }
   slot:0, conn_id:2000, flow_id:1, crypto map:N/R
   no sa timing (manual-keyed)
   replay detection support:N

inbound PCP SAs:

outbound ESP SAs:

outbound AH SAs:
 spi:0x3E8(1000)
   transform:ah-md5-hmac ,
   in use settings ={Transport, }
   slot:0, conn_id:2001, flow_id:2, crypto map:N/R
   no sa timing (manual-keyed)
   replay detection support:N

outbound PCP SAs:
```

What to Do Next

For output examples of the commands used to verify OSPF for IPv6 configuration and operation, refer to the *IPv6 for Cisco IOS Command Reference*.

Configuration Examples for Implementing OSPF for IPv6

This section provides the following configuration examples:

Enabling OSPF for IPv6 on an Interface Configuration: Example

The following example configures an OSPF routing process 109 to run on the interface and puts it in area 1:

```
ipv6 ospf 109 area 1
```

Defining an OSPF for IPv6 Area Range: Example

The following example specifies an OSPF for IPv6 area range:

```
interface Ethernet7/0
 ipv6 address 2001:0DB8:0:0:7::/64 eui-64
 ipv6 enable
 ipv6 ospf 1 area 1
!
interface Ethernet8/0
```

```
 ipv6 address 2001:0DB8:0:0:8::/64 eui-64
 ipv6 enable
 ipv6 ospf 1 area 1
!
interface Ethernet9/0
 ipv6 address 2001:0DB8:0:0:9::/64 eui-64
 ipv6 enable
 ipv6 ospf 1 area 1
!
ipv6 router ospf 1
 router-id 10.11.11.1
 area 1 range 2001:0DB8::/48
```

Defining Authentication on an Interface: Example

The following example defines authentication on the Ethernet 0/0 interface:

```
interface Ethernet0/0
 ipv6 enable
 ipv6 ospf 1 area 0
 ipv6 ospf authentication ipsec spi 500 md5 1234567890ABCDEF1234567890ABCDEF

interface Ethernet0/0
 ipv6 enable
 ipv6 ospf authentication null
 ipv6 ospf 1 area 0
```

Defining Authentication in an OSPF Area: Example

The following example defines authentication on OSPF area 0:

```
ipv6 router ospf 1
 router-id 11.11.11.1
 area 0 authentication ipsec spi 1000 md5 1234567890ABCDEF1234567890ABCDEF
```

Configuring NBMA Interfaces Configuration: Example

The following example configures an OSPF neighboring router with the IPv6 address of FE80::A8BB:CCFF:FE00:C01.

```
interface serial 0
 ipv6 enable
 ipv6 ospf 1 area 0
 encapsulation frame-relay
 frame-relay map ipv6 FE80::A8BB:CCFF:FE00:C01 120
 ipv6 ospf neighbor FE80::A8BB:CCFF:FE00:C0
```

Forcing SPF Configuration: Example

The following example triggers SPF to redo the SPF and repopulate the routing tables:

```
clear ipv6 ospf force-spf
```

Additional References

The following sections provide additional references related to the Implementing OSPF for IPv6 feature.

Related Documents

Related Topic	Document Title
OSPF for IPv4 tasks	"OSPF" chapter of the *Cisco IOS IP Routing Protocols Configuration Guide*, Release 12.4
OSPF for IPv4 commands	*Cisco IOS IP Routing Protocols Command Reference*, Release 12.4
Configuring a router ID in OSPF	"OSPF" chapter of the *Cisco IOS IP Routing Protocols Configuration Guide*, Release 12.4 *Cisco IOS IP Routing Protocols Command Reference*, Release 12.4
OSPF for IPv6 commands	*Cisco IOS IPv6 Command Reference*
IPv6 supported feature list	*Start Here: Cisco IOS Software Release Specifics for IPv6 Features*
Getting IPv6 started	*Implementing Addressing and Basic Connectivity for IPv6*
IPSec for IPv6	*Implementing IPSec for IPv6 Security*
Security configuration tasks (IPv4)	*Cisco IOS Security Configuration Guide*
Security commands: complete command syntax, command mode, defaults, usage guidelines, and examples (IPv4)	*Cisco IOS Security Command Reference*
IPv4 configuration and command reference information	Cisco IOS Configuration Guides and Command References, Release 12.4

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
• CISCO-IETF-IP-FORWARD-MIB • CISCO-IETF-IP-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2401	*Security Architecture for the Internet Protocol*
RFC 2402	*IP Authentication Header*
RFC 2406	*IP Encapsulating Security Payload (ESP)*
RFC 2740	*OSPF for IPv6*
RFC 4552	*Authentication/Confidentiality for OSPFv3*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password. If you have a valid service contract but do not have a user ID or password, you can register on Cisco.com.	http://www.cisco.com/techsupport

Feature Information for Implementing OSPF for IPv6

Table 9 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 12.0(24)S or a later release appear in the table.

For information on a feature in this technology that is not documented here, see “Start Here: Cisco IOS Software Release Specifies for IPv6 Features.”

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS and Catalyst OS software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 9 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 9 Feature Information for Mobile IPv6

Feature Name	Releases	Feature Information
IPv6 routing: OSPF for IPv6 (OSPFv3)	12.0(24)S, 12.2(18)S, 12.2(28)SB, 12.2(33)SRA, 12.2(15)T, 12.3, 12.3(2)T, 12.4, 12.4(2)T	OSPF version 3 for IPv6 expands on OSPF version 2 to provide support for IPv6 routing prefixes and the larger size of IPv6 addresses. This entire document provides information about this feature.
IPv6 routing: LSA types in OSPF for IPv6	12.0(24)S, 12.2(18)S, 12.2(28)SB, 12.2(15)T, 12.3, 12.3(2)T, 12.4, 12.4(2)T	A router’s collection of LSA data is stored in a link-state database. The contents of the database, when subjected to the Dijkstra algorithm, result in the creation of the OSPF routing table. The following sections provide information about this feature: • How OSPF for IPv6 Works, page 188 • LSA Types for IPv6, page 189
IPv6 routing: NBMA interfaces in OSPF for IPv6	12.0(24)S, 12.2(18)S, 12.2(28)SB, 12.2(15)T, 12.3, 12.3(2)T, 12.4, 12.4(2)T	On NBMA networks, the DR or backup DR performs the LSA flooding. The following sections provide information about this feature: • NBMA in OSPF for IPv6, page 190 • Configuring NBMA Interfaces, page 200
IPv6 routing: Force SPF in OSPF for IPv6	12.0(24)S, 12.2(18)S, 12.2(28)SB, 12.2(15)T, 12.3, 12.3(2)T, 12.4, 12.4(2)T	This feature enables the OSPF database to be cleared and repopulated, and then the SPF algorithm is performed. The following sections provide information about this feature: • Force SPF in OSPF for IPv6, page 191 • Forcing an SPF Calculation, page 201

Table 9 Feature Information for Mobile IPv6

Feature Name	Releases	Feature Information
IPv6 routing: Load balancing in OSPF for IPv6	12.0(24)S, 12.2(18)S, 12.2(28)SB, 12.2(15)T, 12.3, 12.3(2)T, 12.4, 12.4(2)T	OSPF for IPv6 performs load balancing automatically. The following sections provide information about this feature: • Load Balancing in OSPF for IPv6, page 191
IPv6 routing: OSPF for IPv6 authentication support with IPSec	12.3(4)T, 12.4, 12.4(2)T	OSPF for IPv6 uses the IPSec secure socket API to add authentication to OSPF for IPv6 packets. The following sections provide information about this feature: • OSPF for IPv6 Authentication Support with IPSec, page 192 • Configuring IPSec on OSPF for IPv6, page 195 • Defining Authentication on an Interface, page 195 • Defining Authentication in an OSPF Area, page 197
IPv6 routing: OSPF IPv6 (OSPFv3) IPSec ESP encryption and authentication	12.4(9)T	IPv6 ESP extension headers can be used to provide authentication and confidentiality to OSPF for IPv6. The following sections provide information about this feature: • Restrictions for Implementing OSPF for IPv6, page 188 • OSPF for IPv6 Authentication Support with IPSec, page 192 • Defining Encryption on an Interface, page 196 • Defining Encryption in an OSPF Area, page 198 • Defining Authentication and Encryption for a Virtual Link in an OSPF Area, page 199