



CHAPTER 21

Configuring OSPF

This chapter describes how to configure the ASA to route data, perform authentication, and redistribute routing information, using the Open Shortest Path First (OSPF) routing protocol.

This chapter includes the following sections:

- [Overview, page 21-1](#)
- [Licensing Requirements for OSPF, page 21-2](#)
- [Guidelines and Limitations, page 21-3](#)
- [Enabling OSPF, page 21-3](#)
- [Customizing OSPF, page 21-4](#)
- [Monitoring OSPF, page 21-15](#)
- [Configuration Example for OSPF, page 21-16](#)
- [Feature History for OSPF, page 21-17](#)
- [Additional References, page 21-17](#)

Overview

OSPF is an interior gateway routing protocol that uses link states rather than distance vectors for path selection. OSPF propagates link-state advertisements rather than routing table updates. Because only LSAs are exchanged instead of the entire routing tables, OSPF networks converge more quickly than RIP networks.

OSPF uses a link-state algorithm to build and calculate the shortest path to all known destinations. Each router in an OSPF area contains an identical link-state database, which is a list of each of the router usable interfaces and reachable neighbors.

The advantages of OSPF over RIP include the following:

- OSPF link-state database updates are sent less frequently than RIP updates, and the link-state database is updated instantly rather than gradually as stale information is timed out.
- Routing decisions are based on cost, which is an indication of the overhead required to send packets across a certain interface. The ASA calculates the cost of an interface based on link bandwidth rather than the number of hops to the destination. The cost can be configured to specify preferred paths.

The disadvantage of shortest path first algorithms is that they require a lot of CPU cycles and memory.

The ASA can run two processes of OSPF protocol simultaneously, on different sets of interfaces. You might want to run two processes if you have interfaces that use the same IP addresses (NAT allows these interfaces to coexist, but OSPF does not allow overlapping addresses). Or you might want to run one process on the inside, and another on the outside, and redistribute a subset of routes between the two processes. Similarly, you might need to segregate private addresses from public addresses.

You can redistribute routes into an OSPF routing process from another OSPF routing process, a RIP routing process, or from static and connected routes configured on OSPF-enabled interfaces.

The ASA supports the following OSPF features:

- Support of intra-area, interarea, and external (Type I and Type II) routes.
- Support of a virtual link.
- OSPF LSA flooding.
- Authentication to OSPF packets (both password and MD5 authentication).
- Support for configuring the ASA as a designated router or a designated backup router. The ASA also can be set up as an ABR.
- Support for stub areas and not-so-stubby-areas.

Area boundary router type-3 LSA filtering.

OSPF supports MD5 and clear text neighbor authentication. Authentication should be used with all routing protocols when possible because route redistribution between OSPF and other protocols (like RIP) can potentially be used by attackers to subvert routing information.

If NAT is used, if OSPF is operating on public and private areas, and if address filtering is required, then you need to run two OSPF processes—one process for the public areas and one for the private areas.

A router that has interfaces in multiple areas is called an Area Border Router (ABR). A router that acts as a gateway to redistribute traffic between routers using OSPF and routers using other routing protocols is called an Autonomous System Boundary Router (ASBR).

An ABR uses LSAs to send information about available routes to other OSPF routers. Using ABR type 3 LSA filtering, you can have separate private and public areas with the ASA acting as an ABR. Type 3 LSAs (inter-area routes) can be filtered from one area to other. This lets you use NAT and OSPF together without advertising private networks.

**Note**

Only type 3 LSAs can be filtered. If you configure the ASA as an ASBR in a private network, it will send type 5 LSAs describing private networks, which will get flooded to the entire AS including public areas.

If NAT is employed but OSPF is only running in public areas, then routes to public networks can be redistributed inside the private network, either as default or type 5 AS External LSAs. However, you need to configure static routes for the private networks protected by the ASA. Also, you should not mix public and private networks on the same ASA interface.

You can have two OSPF routing processes, one RIP routing process, and one EIGRP routing process running on the ASA at the same time.

Licensing Requirements for OSPF

Model	License Requirement
All models	Base License.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature:

Context Mode Guidelines

Supported in single context mode.

Firewall Mode Guidelines

Supported in routed firewall mode only. Transparent mode is not supported.

IPv6 Guidelines

Does not support IPv6.

Configuring OSPF

This section explains how to enable and restart the OSPF process on your system. After enabling see the section, to learn how to customize the OSPF process on your system.

- [Enabling OSPF, page 21-3](#)
- [Restarting the OSPF Process, page 21-4](#)

Enabling OSPF

To enable OSPF, you need to create an OSPF routing process, specify the range of IP addresses associated with the routing process, then assign area IDs associated with that range of IP addresses.

To enable OSPF, perform the following detailed steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	network <i>ip_address mask area area_id</i> Example: hostname(config)# router ospf 2 hostname(config-router)# network 10.0.0.0 255.0.0.0 area 0	This step defines the IP addresses on which OSPF runs and to define the area ID for that interface.

Restarting the OSPF Process

This step allows you to remove the entire OSPF configuration you have enabled. Once this is cleared, you must reconfigure OSPF again using the **router ospf** command, perform the following step:

Command	Purpose
clear ospf <i>pid</i> { process redistribution neighbors [neighbor [<i>neighbor-interface</i>] [<i>neighbor-id</i>]]} Example: hostname(config)# clear ospf	This remove entire OSPF configuration you have enabled. Once this is cleared, you must reconfigure OSPF again using the router ospf command.

Customizing OSPF

This section explains how to customize the OSPF process and includes the following topics:

- [Redistributing Routes Into OSPF, page 21-5](#)
- [Generating a Default Route, page 21-6](#)
- [Configuring OSPF Interface Parameters, page 21-8](#)
- [Configuring Route Summarization Between OSPF Areas, page 21-8](#)
- [Configuring OSPF Interface Parameters, page 21-8](#)
- [Configuring OSPF Area Parameters, page 21-11](#)
- [Configuring OSPF NSSA, page 21-12](#)
- [Configuring Route Calculation Timers, page 21-13](#)
- [Defining Static OSPF Neighbors, page 21-13](#)
- [Logging Neighbors Going Up or Down, page 21-14](#)

Redistributing Routes Into OSPF

The ASA can control the redistribution of routes between OSPF routing processes. The ASA matches and changes routes according to settings in the **redistribute** command or by using a route map.

If you want to redistribute a route by defining which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process, you must first generate a default map and then define a route map.



Note

(Optional) Create a route-map to further define which routes from the specified routing protocol are redistributed in to the OSPF routing process. See [Chapter 20, “Defining Route Maps.”](#) Also, see the [“Generating a Default Route”](#) section on page 21-6 for another use for route maps.

To redistribute static, connected, RIP, or OSPF routes into an OSPF process, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i>	This creates an OSPF routing process, and the user enters router configuration mode for the OSPF process you want to redistribute.
	Example: hostname(config)# router ospf 2	The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	Do one of the following to redistribute the selected route type into the OSPF routing process:	
	redistribute connected [[metric <i>metric-value</i>] [metric-type { type-1 type-2 }] [tag <i>tag_value</i>] [subnets] [route-map <i>map_name</i>]	This step redistributes connected routes into the OSPF routing process
	redistribute static [metric <i>metric-value</i>] [metric-type { type-1 type-2 }] [tag <i>tag_value</i>] [subnets] [route-map <i>map_name</i>] Example: hostname(config)# redistribute static	This step redistribute static routes into the OSPF routing process.

Command	Purpose
<pre> redistribute ospf <i>pid</i> [match {<i>internal</i> <i>external</i> [<i>1</i> <i>2</i>] <i>nssa-external</i> [<i>1</i> <i>2</i>]}] [metric <i>metric-value</i>] [metric-type {<i>type-1</i> <i>type-2</i>}] [tag <i>tag_value</i>] [subnets] [route-map <i>map_name</i>] </pre> Example: <pre> hostname(config)# route-map 1-to-2 permit hostname(config-route-map)# match metric 1 hostname(config-route-map)# set metric 5 hostname(config-route-map)# set metric-type type-1 hostname(config-route-map)# router ospf 2 hostname(config-router)# redistribute ospf 1 route-map 1-to-2 </pre>	This step allows you to redistribute routes from an OSPF routing process into another OSPF routing process. You can either use the match options in this command to match and set route properties, or you can use a route map. The subnet option does not have equivalents in the route-map command. If you use both a route map and match options in the redistribute command, then they must match. This example shows route redistribution from OSPF process 1 into OSPF process 2 by matching routes with a metric equal to 1. The ASA redistributes these routes as external LSAs with a metric of 5, metric type of Type 1.
<pre> redistribute rip [metric <i>metric-value</i>] [metric-type {<i>type-1</i> <i>type-2</i>}] [tag <i>tag_value</i>] [subnets] [route-map <i>map_name</i>] </pre> Example: <pre> hostname(config)# redistribute rip 25 </pre>	This step allows you to redistribute routes from a RIP routing process into the OSPF routing process.
<pre> redistribute eigrp as-num [metric <i>metric-value</i>] [metric-type {<i>type-1</i> <i>type-2</i>}] [tag <i>tag_value</i>] [subnets] [route-map <i>map_name</i>] </pre> Example: <pre> hostname(config)# redistribute eigrp 2 </pre>	This step allows you to redistribute routes from an EIGRP routing process into the OSPF routing process.

Generating a Default Route

You can force an autonomous system boundary router to generate a default route into an OSPF routing domain. Whenever you specifically configure redistribution of routes into an OSPF routing domain, the router automatically becomes an autonomous system boundary router. However, an autonomous system boundary router does not by default generate a default route into the OSPF routing domain.

To generate a default route, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	default-information originate [always] [metric <i>metric-value</i>] [metric-type {1 2}] [route-map <i>map-name</i>] Example: hostname(config-router)# default-information originate always	This step forces the autonomous system boundary router to generate a default route.

Configuring Route Summarization When Redistributing Routes into OSPF

When routes from other protocols are redistributed into OSPF, each route is advertised individually in an external LSA. However, you can configure the ASA to advertise a single route for all the redistributed routes that are covered by a specified network address and mask. This configuration decreases the size of the OSPF link-state database.

To configure the software advertisement on one summary route for all redistributed routes covered by a network address and mask, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 1	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	summary-address <i>ip_address mask</i> [not-advertise] [tag <i>tag</i>] Example: hostname(config)# router ospf 1 hostname(config-router)# summary-address 10.1.0.0 255.255.0.0	This step sets the summary address. In this example, the summary address 10.1.0.0 includes address 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the address 10.1.0.0 is advertised in an external link-state advertisement



Note

OSPF does not support **summary-address 0.0.0.0 0.0.0.0**.

Configuring Route Summarization Between OSPF Areas

Route summarization is the consolidation of advertised addresses. This feature causes a single summary route to be advertised to other areas by an area boundary router. In OSPF, an area boundary router advertises networks in one area into another area. If the network numbers in an area are assigned in a way such that they are contiguous, you can configure the area boundary router to advertise a summary route that covers all the individual networks within the area that fall into the specified range.

To define an address range for route summarization, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 1	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	area <i>area-id</i> range <i>ip-address mask</i> [advertise not-advertise] Example: hostname(config)# router ospf 1 hostname(config-router)# area 17 range 12.1.0.0 255.255.0.0	This step sets the address range. In this example, the address range is set between OSPF areas.

Configuring OSPF Interface Parameters

You can alter some interface-specific OSPF parameters as necessary. You are not required to alter any of these parameters, but the following interface parameters must be consistent across all routers in an attached network: **ospf hello-interval**, **ospf dead-interval**, and **ospf authentication-key**. Be sure that if you configure any of these parameters, the configurations for all routers on your network have compatible values.

To configure OSPF interface parameters, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for the OSPF process you want to redistribute. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	network <i>ip_address mask area area_id</i> Example: hostname(config)# router ospf 2 hostname(config-router)# network 10.0.0.0 255.0.0.0 area 0	This step defines the IP addresses on which OSPF runs and to define the area ID for that interface.
Step 3	hostname(config)# interface <i>interface_name</i> Example: hostname(config)# interface <i>my_interface</i>	This allows you to enter interface configuration mode.
Step 4	Do one of the following to configure optional OSPF interface parameters:	
	ospf authentication [message-digest null] Example: hostname(config-interface)# ospf authentication message-digest	This specifies the authentication type for an interface.
	ospf authentication-key <i>key</i> Example: hostname(config-interface)# ospf authentication-key cisco	This allows you to assign a password to be used by neighboring OSPF routers on a network segment that is using the OSPF simple password authentication. The <i>key</i> can be any continuous string of characters up to 8 bytes in length. The password created by this command is used as a key that is inserted directly into the OSPF header when the ASA software originates routing protocol packets. A separate password can be assigned to each network on a per-interface basis. All neighboring routers on the same network must have the same password to be able to exchange OSPF information.
	ospf cost <i>cost</i> Example: hostname(config-interface)# ospf cost 20	This allows you to explicitly specify the cost of sending a packet on an OSPF interface. The <i>cost</i> is an integer from 1 to 65535. In this example, the cost is set to 20.
	ospf dead-interval <i>seconds</i> Example: hostname(config-interface)# ospf dead-interval 40	This allows you to set the number of seconds that a device must wait before it declares a neighbor OSPF router down because it has not received a hello packet. The value must be the same for all nodes on the network. In this example, the dead-interval is set to 40.

Command	Purpose
ospf hello-interval <i>seconds</i> Example: hostname(config-interface)# ospf hello-interval 10	This allows you to specify the length of time between the hello packets that the ASA sends on an OSPF interface. The value must be the same for all nodes on the network. In this example, the hello-interval is set to 10.
ospf message-digest-key <i>key_id md5 key</i> Example: hostname(config-interface)# ospf message-digest-key 1 md5 cisco	This enables OSPF MD5 authentication. The following values can be set: <i>key_id</i>—An identifier in the range from 1 to 255. <i>key</i>—Alphanumeric password of up to 16 bytes. Usually, one key per interface is used to generate authentication information when sending packets and to authenticate incoming packets. The same key identifier on the neighbor router must have the same key value. We recommend that you not keep more than one key per interface. Every time you add a new key, you should remove the old key to prevent the local system from continuing to communicate with a hostile system that knows the old key. Removing the old key also reduces overhead during rollover.
ospf priority <i>number_value</i> Example: hostname(config-interface)# ospf priority 20	This allows you to set the priority to help determine the OSPF designated router for a network. The <i>number_value</i> is between 0 to 255. In this example, the priority number value is set to 20.
ospf retransmit-interval <i>seconds</i> Example: hostname(config-interface)# ospf retransmit-interval <i>seconds</i>	This allows you to specify the number of seconds between LSA retransmissions for adjacencies belonging to an OSPF interface. The value for <i>seconds</i> must be greater than the expected round-trip delay between any two routers on the attached network. The range is from 1 to 65535 seconds. The default value is 5 seconds. In this example, the retransmit-interval value is set to 15.
ospf transmit-delay <i>seconds</i> Example: hostname(config-interface)# ospf transmit-delay 5	Sets the estimated number of seconds required to send a link-state update packet on an OSPF interface. The <i>seconds</i> value is from 1 to 65535 seconds. The default value is 1 second. In this example, the transmit-delay is 5 seconds.
ospf network point-to-point non-broadcast Example: hostname(config-interface)# ospf network point-to-point non-broadcast	Specifies the interface as a point-to-point, non-broadcast network. When you designate an interface as point-to-point, non-broadcast, you must manually define the OSPF neighbor; dynamic neighbor discover is not possible. See Defining Static OSPF Neighbors, page 21-13, for more information. Additionally, you can only define one OSPF neighbor on that interface.

Configuring OSPF Area Parameters

You can configure several area parameters. These area parameters (shown in the following task table) include setting authentication, defining stub areas, and assigning specific costs to the default summary route. Authentication provides password-based protection against unauthorized access to an area.

Stub areas are areas into which information on external routes is not sent. Instead, there is a default external route generated by the ABR, into the stub area for destinations outside the autonomous system. To take advantage of the OSPF stub area support, default routing must be used in the stub area. To further reduce the number of LSAs sent into a stub area, you can configure the **no-summary** keyword of the **area stub** command on the ABR to prevent it from sending summary link advertisement (LSA Type 3) into the stub area.

To specify area parameters for your network, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i>	This creates an OSPF routing process, and the user enters router configuration mode for the OSPF process you want to redistribute.
	Example: hostname(config)# router ospf 2	The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	Do one of the following to configure optional OSPF area parameters:	
	area <i>area-id</i> authentication	This step enables authentication for an OSPF area.
	Example: hostname(config-router)# area 0 authentication	
	area <i>area-id</i> authentication message-digest	This step enables MD5 authentication for an OSPF area.
	Example: hostname(config-router)# area 0 authentication message-digest	
	area <i>area-id</i> stub [no-summary]	This defines an area to be a stub area.
	Example: hostname(config-router)# area 17 stub	
	area <i>area-id</i> default-cost <i>cost</i>	This step assigns a specific cost to the default summary route used for the stub area.
	Example: hostname(config-router)# area 17 default-cost 20	The <i>cost</i> is an integer from 1 to 65535. The default value is 1.

Configuring OSPF NSSA

The OSPF implementation of an NSSA is similar to an OSPF stub area. NSSA does not flood type 5 external LSAs from the core into the area, but it can import autonomous system external routes in a limited way within the area.

NSSA imports Type 7 autonomous system external routes within an NSSA area by redistribution. These Type 7 LSAs are translated into Type 5 LSAs by NSSA ABRs, which are flooded throughout the whole routing domain. Summarization and filtering are supported during the translation.

You can simplify administration if you are an ISP or a network administrator that must connect a central site using OSPF to a remote site that is using a different routing protocol using NSSA.

Before the implementation of NSSA, the connection between the corporate site border router and the remote router could not be run as an OSPF stub area because routes for the remote site could not be redistributed into the stub area, and two routing protocols needed to be maintained. A simple protocol such as RIP was usually run and handled the redistribution. With NSSA, you can extend OSPF to cover the remote connection by defining the area between the corporate router and the remote router as an NSSA.

Before you use this feature, consider these guidelines:

- You can set a Type 7 default route that can be used to reach external destinations. When configured, the router generates a Type 7 default into the NSSA or the NSSA area boundary router.
- Every router within the same area must agree that the area is NSSA; otherwise, the routers will not be able to communicate.

To specify area parameters for your network as needed to configure OSPF NSSA, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for the OSPF process you want to redistribute. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	Do one of the following to configure optional OSPF NSSA parameters:	

Command	Purpose
area <i>area-id</i> nssa [no-redistribution] [default-information-originate] Example: hostname(config-router)# area 0 nssa	This step defines an NSSA area.
summary-address <i>ip_address mask</i> [not-advertise] [tag tag] Example: hostname(config)# router ospf 1 hostname(config-router)# summary-address 10.1.0.0 255.255.0.0	This step sets the summary address and helps reduce the size of the routing table. Using this command for OSPF causes an OSPF ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by the address. In this example, the summary address 10.1.0.0 includes address 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the address 10.1.0.0 is advertised in an external link-state advertisement

**Note**

OSPF does not support **summary-address 0.0.0.0 0.0.0.0**.

Defining Static OSPF Neighbors

You need to define static OSPF neighbors to advertise OSPF routes over a point-to-point, non-broadcast network. This lets you broadcast OSPF advertisements across an existing VPN connection without having to encapsulate the advertisements in a GRE tunnel.

Before you begin, you must create a static route to the OSPF neighbor. See the chapter, ‘[Configuring Static and Default Routes](#)’ for more information about creating static routes.

To define a static OSPF neighbor, perform the following tasks:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	neighbor <i>addr</i> [interface <i>if_name</i>] Example: hostname(config-router)# neighbor 255.255.0.0 [interface <i>my_interface</i>]	This step defines the OSPF neighborhood. The <i>addr</i> argument is the IP address of the OSPF neighbor. The <i>if_name</i> is the interface used to communicate with the neighbor. If the OSPF neighbor is not on the same network as any of the directly-connected interfaces, you must specify the interface.

Configuring Route Calculation Timers

You can configure the delay time between when OSPF receives a topology change and when it starts an SPF calculation. You also can configure the hold time between two consecutive SPF calculations.

To configure route calculation timers, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	timers spf <i>spf-delay</i> <i>spf-holdtime</i> Example: hostname(config-router)# timers spf 10 120	This step configure the route calculation times. The <i>spf-delay</i> is the delay time (in seconds) between when OSPF receives a topology change and when it starts an SPF calculation. It can be an integer from 0 to 65535. The default time is 5 seconds. A value of 0 means that there is no delay; that is, the SPF calculation is started immediately. The <i>spf-holdtime</i> is the minimum time (in seconds) between two consecutive SPF calculations. It can be an integer from 0 to 65535. The default time is 10 seconds. A value of 0 means that there is no delay; that is, two SPF calculations can be done, one immediately after the other.

Logging Neighbors Going Up or Down

By default, the system sends a system message when an OSPF neighbor goes up or down.

Configure this command if you want to know about OSPF neighbors going up or down without turning on the **debug ospf adjacency** command. The **log-adj-changes** router configuration command provides a higher level view of the peer relationship with less output. Configure **log-adj-changes detail** if you want to see messages for each state change.

To log neighbors going up or down, perform the following steps:

Detailed Steps

	Command	Purpose
Step 1	router ospf <i>process_id</i> Example: hostname(config)# router ospf 2	This creates an OSPF routing process, and the user enters router configuration mode for this OSPF process. The <i>process_id</i> is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.
Step 2	log-adj-changes [detail] Example: hostname(config-router)# log-adj-changes [detail]	This step configures logging for neighbors going up or down.

**Note**

Logging must be enabled for the the neighbor up/down messages to be sent.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases. You can also use the information provided to determine resource utilization and solve network problems. You can also display information about node reachability and discover the routing path that your device packets are taking through the network.

To monitor or display various OSPF routing statistics, perform one of the following tasks:

Command	Purpose
show ospf [<i>process-id</i> [<i>area-id</i>]]	Displays general information about OSPF routing processes.
show ospf border-routers	Displays the internal OSPF routing table entries to the ABR and ASBR.
show ospf [<i>process-id</i> [<i>area-id</i>]] database	Displays lists of information related to the OSPF database for a specific router.
show ospf flood-list <i>if-name</i>	Displays a list of LSAs waiting to be flooded over an interface (to observe OSPF packet pacing). OSPF update packets are automatically paced so they are not sent less than 33 milliseconds apart. Without pacing, some update packets could get lost in situations where the link is slow, a neighbor could not receive the updates quickly enough, or the router could run out of buffer space. For example, without pacing packets might be dropped if either of the following topologies exist: • A fast router is connected to a slower router over a point-to-point link. • During flooding, several neighbors send updates to a single router at the same time. Pacing is also used between resends to increase efficiency and minimize lost retransmissions. You also can display the LSAs waiting to be sent out an interface. The benefit of the pacing is that OSPF update and retransmission packets are sent more efficiently. There are no configuration tasks for this feature; it occurs automatically
show ospf interface [<i>if_name</i>]	Displays OSPF-related interface information.
show ospf neighbor [<i>interface-name</i>] [<i>neighbor-id</i>] [detail]	Displays OSPF neighbor information on a per-interface basis.
show ospf request-list <i>neighbor if_name</i>	Displays a list of all LSAs requested by a router.

Command	Purpose
show ospf retransmission-list <i>neighbor if_name</i>	Displays a list of all LSAs waiting to be resent.
show ospf [<i>process-id</i>] summary-address	Displays a list of all summary address redistribution information configured under an OSPF process.
show ospf [<i>process-id</i>] virtual-links	Displays OSPF-related virtual links information.

Configuration Example for OSPF

The following example shows how to enable and configure OSPF with various optional processes:

Step 1 Enable OSPF.

```
hostname(config)# router ospf 2
hostname(config-router)# network 10.0.0.0 255.0.0.0 area 0
```

Step 2 Redistribute routes from one OSPF process to another OSPF process (optional):

```
hostname(config)# route-map 1-to-2 permit
hostname(config-route-map)# match metric 1
hostname(config-route-map)# set metric 5
hostname(config-route-map)# set metric-type type-1
hostname(config-route-map)# router ospf 2
hostname(config-router)# redistribute ospf 1 route-map 1-to-2
```

Step 3 Configure OSPF interface parameters (optional):

```
hostname(config)# router ospf 2
hostname(config-router)# network 2.0.0.0 255.0.0.0 area 0
hostname(config-router)# interface inside
hostname(config-interface)# ospf cost 20
hostname(config-interface)# ospf retransmit-interval 15
hostname(config-interface)# ospf transmit-delay 10
hostname(config-interface)# ospf priority 20
hostname(config-interface)# ospf hello-interval 10
hostname(config-interface)# ospf dead-interval 40
hostname(config-interface)# ospf authentication-key cisco
hostname(config-interface)# ospf message-digest-key 1 md5 cisco
hostname(config-interface)# ospf authentication message-digest
```

Step 4 Configure OSPF area parameters (optional):

```
hostname(config)# router ospf 2
hostname(config-router)# area 0 authentication
hostname(config-router)# area 0 authentication message-digest
hostname(config-router)# area 17 stub
hostname(config-router)# area 17 default-cost 20
```

Step 5 Configure the route calculation timers and show the log neighbor up/down messages (optional):

```
hostname(config-router)# timers spf 10 120
hostname(config-router)# log-adj-changes [detail]
```

Step 6 Restart the OSPF process .

```
hostname(config)# clear ospf pid {process | redistribution | counters
[neighbor [neighbor-interface] [neighbor-id]]}
```

Step 7 Show the results of your OSPF configuration (optional):

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

```
hostname(config)# show ospf

Routing Process "ospf 2" with ID 20.1.89.2 and Domain ID 0.0.0.2
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 5. Checksum Sum 0x 26da6
Number of opaque AS LSA 0. Checksum Sum 0x      0
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
  Area BACKBONE(0)
    Number of interfaces in this area is 1
    Area has no authentication
    SPF algorithm executed 2 times
    Area ranges are
    Number of LSA 5. Checksum Sum 0x 209a3
    Number of opaque link LSA 0. Checksum Sum 0x      0
    Number of DCbitless LSA 0
    Number of indication LSA 0
    Number of DoNotAge LSA 0
    Flood list length 0
```

Feature History for OSPF

Table 21-1 lists the release history for this feature.

Table 21-1 Feature History for OSPF

Feature Name	Releases	Feature Information
router ospf	7.0	route data, perform authentication, redistribute and monitor routing information, using the Open Shortest Path First (OSPF) routing protocol.

Additional References

For additional information related to routing, see the following:

- [Related Documents, page 21-18](#)

Related Documents

Related Topic	Document Title
Routing Overview	Information About Routing
How to configure EIGRP	Configuring EIGRP
How to configure RIP	Configuring RIP
How to configure a static or default route	Configuring Static and Default Routes
How to configure a route map	Defining Route Maps
How to configure multicast routing	Configuring Multicast Routing