CHAPTER 10

Configuring OSPFv3 for a MANET

This chapter provides the following major sections for configuring OSPFv3 in a Mobile Ad-hoc Network (MANET):

OSPFv3 for MANET

Configuring OSPFv3 for a MANET has similar requirements to many traditional OSPFv3 configurations. The primary difference is to configure the network type of OSPFv3 as a MANET. To optimize the use of OSPFv3 with MANETs, Cisco IOS implements extensions to OSPFv3 as defined in *IETF RFC 5820*. The result is a well-understood routing protocol designed for a constantly changing network topology constrained by limited bandwidth.

This is accomplished in several ways:

- Radio Aware Routing (RAR): Provides tight coupling of OSPFv3 with cooperative radios (fast convergence and re-convergence through neighbor-presence indicators). Determines accurate, real-time, link-metric costs.
- Incremental Hello: Minimizes OSPFv3 packet size.
- Caching Multicast Link-State Advertisements (LSAs): Minimizes OSPFv3 packet transmissions.
- Optimized Flooding (Overlapping Relay): Minimizes the number of flooded LSAs.
- Selective Peering: Reduces OSPFv3 network overhead by limiting redundant full-peering adjacencies.

Cooperative Radios

While non-cooperative radios are supported, OSPFv3 in a MANET operates best when used with cooperative radios, which is a configuration requiring Virtual Multipoint Interfaces (VMIs). See Chapter 5, "Introduction to Radio Aware Routing and MANET" for detailed procedures.

Note This document defines a Cooperative radio as a radio containing the firmware and software required to support RAR-based flows.

Initial Configuration Procedures

Configuring OSPFv3 for a MANET begins with the following tasks:

1. Enabling IPv6 Routing, page 10-2
2. Enabling IPv6 on the Interface, page 10-3
3. Configuring the OSPFv3 Process, page 10-4
4. Configuring the Interface for OSPFv3 MANETs, page 10-5

Enabling IPv6 Routing

This task enables IPv6 packet forwarding and IPv6 routing, both disabled by default.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 unicast-routing**
4. **exit**

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 unicast-routing`
	Example:
	`Router(config)# ipv6 unicast-routing`
		Enables the forwarding of IPv6 unicast datagrams.

Step 4	`exit`
	Example:
	`Router(config)# exit`
		Exits global configuration mode and returns the router to privileged EXEC mode.

Enabling IPv6 on the Interface

This task enables IPv6 on an interface—a prerequisite to configuring OSPFv3 on the interface. IPv6 is disabled by default.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** [*type number*]
4. **ipv6 enable**
5. **exit**

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.

	Command or Action	Purpose

Step 4	`ipv6 enable`
	Example:
	`Router(config-if)# ipv6 enable`
		Enables IPv6 processing on an interface that has not been configured with an explicit IPv6 address.

Step 5	`exit`
	Example:
	`Router(config-if)# exit`
		Exits global configuration mode and returns the router to privileged EXEC mode.

Configuring the OSPFv3 Process

This task configures the OSPFv3 process for IPv6 or IPv4.

Note The commands in this task indicate IPv6. If you want to configure the OSPFv3 process for IPv4 instead, see the detailed steps for examples.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router ospfv3** [*process*-id]
4. **router-id** [*OSPFv3 router-id in IP address format*]
5. **address-family ipv6 unicast**
6. **exit**

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	***router ospfv3*** [*process-id*] **Example:** Router(config)# **router ospfv3 6** **Example for IPv4:** Router(config)# **router ospfv3 4**	Enables OSPFv3 for IPv6 router configuration mode.
Step 4	**router-id** [*OSPFv3 router-id in IP address format*] **Example:** Router(config-rtr)# **router-id 10.1.1.1**	Enables the use of a fixed router ID.
Step 5	**address-family ipv6 unicast** **Example:** Router(config-rtr)# **address-family ipv6 unicast** **Example for IPv4:** Router(config-rtr)# **address-family ipv4 unicast**	Enables the address family for IPv6.
Step 6	**exit** **Example:** Router(config-rtr)# **exit**	Exits global configuration mode and returns the router to privileged EXEC mode.

Configuring the Interface for OSPFv3 MANETs

This configures the OSPFv3 process for IPv6 or IPv4.

Note The commands in this task indicate IPv6. If you want to configure the OSPFv3 process for IPv4 instead, see the detailed steps for examples.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** [*type number*]
4. **ospfv3** [*process-id*] **area** *area-id* **ipv6 [instance** *instance-id*]
5. **ospfv3** [*process-id*] **network manet**
6. **exit**

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 vmi1**	Configures an interface type.
Step 4	**ospfv3** [*process-id*] **area** area-id **ipv6** [**instance** instance-id] **Example:** (Router-if)# **ospfv3 6 area 0 ipv6** **Example for IPv4:** (Router-if)# **ospfv3 6 area 0 ipv4**	Attaches the OSPFv3 process to an interface. **Note** The instance number defaults to 0 for ipv6.
Step 5	**ospfv3** [*process-id*] **network manet** **Example:** Router(config-if)# **ospfv3 6 network manet** **Example for IPv4:** Router(config-if)# **ospfv3 4 network manet**	Configures the OSPFv3 network type to MANET.
Step 6	**exit** **Example:** Router(config-if)# **exit**	Exits global configuration mode and returns the router to privileged EXEC mode.

EXAMPLE

```
version 15.1
!
hostname Router1
!
...
<output truncated>
...
interface Ethernet0/0
 no ip address
 ipv6 enable
 ospfv3 6 network manet
 ospfv3 6 area 0 ipv6
!
interface Ethernet0/1
 ip address 4.4.4.4 255.255.255.0
 ipv6 enable
 ospfv3 4 network manet
 ospfv3 4 area 0 ipv4
shutdown
!
interface Ethernet0/2
 no ip address
 shutdown
!
interface Ethernet0/3
 no ip address
 shutdown
!
ip forward-protocol nd
!
no ip http server
!
router ospfv3 6
 router-id 1.1.1.1
 address-family ipv6 unicast
 log-adjacency-changes
!
router ospfv3 4
 router-id 4.4.4.4
 address-family ipv4 unicast
 log-adjacency-changes
...
<output truncated>
...
end
```

Radio Aware Routing in a MANET

This section describes how to configure OSPFv3 in MANETs for Radio Aware Routing (RAR).

Prerequisites

All radios in OSPFv3 for MANET must be cooperative radios attached to a Virtual Multipoint Interface (VMI).

Before performing the tasks in this section, you must configure OSPFv3 for MANETs on a VMI as described in Chapter 5, "Introduction to Radio Aware Routing and MANET".

Link Metrics

Cooperative radios in the MANET report link-quality metrics, which can include the following information:

- Maximum Data Rate—the theoretical maximum data rate of the radio link, in bytes per second
- Current Data Rate—the current data rate achieved on the link, in bytes per second
- Latency—the transmission delay packets encounter, in milliseconds
- Resources—a percentage (0-100) that can represent the remaining amount of a resource (such as battery power)
- Relative Link Quality—a numeric value (0-100) representing relative quality, with 100 being the highest quality

Fine-Tuning RAR Configurations

You can fine-tune RAR configurations within a MANET by converting the link metrics to OSPFv3 link costs and configuring a hysteresis threshold. Configuring a hysteresis threshold on the resultant link costs helps minimize the propagation of LSAs responding to link-metric changes.

Metrics can be weighted during the configuration process to emphasize or de-emphasize particular characteristics. For example, if throughput is highly important, the metric for Current Data Rate (CDR) could be weighted more heavily into the composite metric. Similarly, a metric that is of no concern can be omitted.

Link metrics can change rapidly, often by very small degrees, which can result in a flood of meaningless routing updates. In a worst case scenario, the network will churn almost continuously as it struggles to react to minor variations in link quality. To alleviate this concern, Cisco provides a tunable dampening mechanism that allows the user to configure threshold values. Any metric change that falls below the threshold is ignored.

A tunable hysteresis mechanism allows users to adjust the threshold to the routing changes that occur when the router receives a signal that a new peer has been discovered, or that an existing peer is unreachable. The tunable metric is weighted and is adjusted dynamically to account for the following characteristics:

- Current and Maximum Bandwidth
- Latency
- Resources

- Hysteresis

Individual weights can be deconfigured and all weights cleared so that the cost is set back to the default value for the interface type. Based on the routing changes that occur, cost can be determined by the application of these metrics.

The dynamic cost metric used for interfaces is computed based on the Layer 2 (L2) feedback to Layer 3 (L3), where the metric calculations are as follows:

OC = maximum-data-rate
S1 = ospfv3 6 dynamic weight throughput (Bandwidth component)
S2 = ospfv3 6 dynamic weight resources (Resources component)
S3 = ospfv3 6 dynamic weight latency (Latency component)
S4 = ospfv3 6 dynamic weight L2 factor (L2 factor component)

Note While the commands and output in this section reflect IPv6 configurations, all examples and commands work for IPv4 as well.

Throughput = (current-data-rate)/(maximum-data-rate)

Router-dynamic cost = OC + (S1) + (S2) + (S3) + (S4)

For a dynamic cost to have the same cost as a default cost, all parameters must equal zero.

Each L2 feedback can contribute a cost in the range of 0 to 65535. To tune down this cost range, use the optional **weight** keyword in conjunction with the **throughput**, **resources**, **latency**, or **L2-factor** keyword. Each of these weights has a default value of 100 percent and can be configured in a range from 0 to 100. When 0 is configured for a specific weight, that weight does not contribute to the OSPFv3 cost.

Because cost components can change rapidly, you may need to dampen the amount of changes in order to reduce network-wide churn. Use the optional **hysteresis** keyword with the **threshold** *threshold-value* keyword and argument to set a cost change threshold. Any cost change below this threshold is ignored.

You can use the **hysteresis** keyword to specify a hysteresis value based on the percentage of change of the currently stored value in the routing table for the peer.

Each time the router receives a new PADQ packet from the radio for a peer, a new cost will be calculated for it. The **hysteresis** keyword specifies the amount of change required before saving the new value.

The hysteresis percent calculated is performed as follows:

If the absolute value of (new_cost - saved_cost) is greater than (hysteresis_percent*saved_cost), then the new_cost will be saved.

Because cost components can change rapidly, it might be necessary to dampen the volume of changes to reduce network-wide churn. The recommended values for S2, S3, and S4 are based on network simulations that may reduce the rate of network changes. The recommended value for S1 is zero to eliminate this variable from the route cost calculation.

While each network might have unique characteristics that require different settings to optimize actual network performance, these are recommended values intended as a starting point for optimizing a OSPFv3 network. Table 10-1 lists the recommended value settings for OSPFv3 cost metrics.

Table 10-1 Recommended Value Settings for OSPFv3 Cost Metrics

Setting	Metric Description	Default Value	Recommended Value
S1	ospfv3 6 dynamic weight throughout	100	0
S2	ospfv3 6 dynamic weight resources	100	29

Table 10-1 Recommended Value Settings for OSPFv3 Cost Metrics

Setting	Metric Description	Default Value	Recommended Value
S3	ospfv3 6 dynamic weight latency	100	29
S4	ospfv3 6 dynamic weight L2-factor	100	29

The overall link cost is computed using the following formula:

$$\text{LinkCost} = \text{OC} + \overset{\textcircled{1}}{\text{BW}}\left(\frac{\text{Throughput\_weight}}{100}\right) + \overset{\textcircled{2}}{\text{Resources}}\left(\frac{\text{Resources\_weight}}{100}\right) + \overset{\textcircled{3}}{\text{Latency}}\left(\frac{\text{Latency\_weight}}{100}\right) + \overset{\textcircled{4}}{\text{L2\_factor}}\left(\frac{\text{L2\_weight}}{100}\right)$$

$$\text{OC} = \left[\frac{(\text{ospf\_reference\_bw})}{(\text{MDR})(1000)}\right] \qquad \boxed{\text{ospf\_reference\_bw} = 10 \wedge 8}$$

$$\textcircled{1}\quad \text{BW} = \frac{(65535)\left(100 - \frac{\text{CDR}}{\text{MDR}}(100)\right)}{100}$$

$$\textcircled{2}\quad \text{Resources} = \frac{(100 - \text{resources})^3 (65535)}{1000000}$$

$$\textcircled{3}\quad \text{Latency} = \text{latency}$$

$$\textcircled{4}\quad \text{L2\_factor} = \frac{(100 - \text{RLQ})(65535)}{100}$$

231048

EXAMPLE

To illustrate these settings, the following example shows how OSPFv3 cost metrics can be defined for a VMI interface with one type of radio:

```
interface vmi1
    ospfv3 6 cost dynamic hysteresis percent 10
    ospfv3 6 cost dynamic weight throughput 0
    ospfv3 6 cost dynamic weight resources 29
    ospfv3 6 cost dynamic weight latency 29
    ospfv3 6 cost dynamic weight L2-factor 29
```

EXAMPLE

The following is an IPv6 example of configuration:

```
version 15.1
!
hostname Router1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
!
ip cef
ipv6 unicast-routing
ipv6 cef
subscriber authorization enable
!
subscriber profile pppoe_group_1
 pppoe service manet_radio
```

```
!
!
multilink bundle-name authenticated
!
no virtual-template subinterface
!
bba-group pppoe pppoe_group_1
 virtual-template 1
 service profile pppoe_group_1
!
interface Ethernet0/0
 no ip address
 shutdown
!
interface Ethernet0/1
 no ip address
 shutdown
!
interface Ethernet0/2
 no ip address
 shutdown
!
interface Ethernet0/3
 no ip address
 shutdown
!
interface Virtual-Template1
 no ip address
 ipv6 enable
 no peer default ip address
 no keepalive
!
interface vmi1
 no ip address
 ipv6 enable
 ospfv3 6 network manet
 ospfv3 6 area 0 ipv6
 physical-interface Ethernet0/0
!
ip forward-protocol nd
!
router ospfv3 6
 router-id 1.1.1.1
 log-adjacency-changes
 address-family ipv6 unicast
 exit-address-family
...
<output truncated>
...
end
```

Selective Peering for Efficiency

Use selective peering to minimize network costs by minimizing each node’s redundant adjacencies. For each OSPFv3 MANET node, you can restrict full-peering rights to the adjacent neighbors that enhance reachability while remaining cost-effective. For each neighbor adjacency calculated to cause excessive link costs, you can use selective peering to keep that neighbor in a 2-way state. This reduces the need for control-plane bandwidth by reducing database exchanges and routing updates.

Note Selective peering does not reduce dataplane connectivity. User traffic will flow over 2-way links when provided with the best path through the network.

Determining Peering Criteria

Upon discovery of each new neighbor within an OSPFv3 MANET node, selective peering determines whether the forming of an adjacency is cost-effective:

- Yes—Form the full-peering adjacency if the neighbor is not in the OSPFv3 link-state database or reachable via the Shortest Path Tree (SPT).
- No—Instead of forming a full-peering adjacency, maintain a 2-way state when the neighbor is in the OSPFv3 link-state database, reachable, and configured with a redundant-path threshold.

Because dynamic topologies can cause a neighbor path redunancy level to fall below the configured threshold, selective peering can change a neighbor 2-way state to full peering.

Link Costs

Selective peering includes link cost as a factor when determining adjacency formation. Ideally, only the links having the lowest costs are granted full-peering adjacency. You can configure OSPFv3 link costs manually, and with cooperative radio interfaces, link costs are obtained directly from the radios through the VMI.

Working with selective peering involves the following tasks:

- Enabling Selective Peering, page 10-12
- Preventing Full Peering over Poor Links, page 10-14
- Fine-Tuning Selective Peering, page 10-15

Enabling Selective Peering

This task explains how to enable OSPFv3 selective peering for IPv6 or IPv4.

Note The commands in this task indicate IPv6. If you want to configure the OSPFv3 process for IPv4 instead, see the detailed steps for examples.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router ospfv3** [*process-id*]
4. **address-family ipv6 unicast**
5. **manet peering selective** [redundancy <*level*>] [*per-interface*]
6. **exit**

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	`router ospfv3` *`[process-id]`*
	Example:
	`Router (config)# router ospfv3 6`
	Example for IPv4:
	`Router(config)# router ospfv3 4`
		Creates OSPFv3 process.

Step 4	`address-family ipv6 unicast`
	Example:
	`Router (config)# address-family ipv6 unicast`
	Example for IPv4:
	`Router(config)# address-family ipv4 unicast`
		Specifies that the OSPFv3 process supports the IPv6 unicast address family.

Step 5	`manet peering selective [redundancy <level>] [per-interface]`
	Example:
	`Router(config-rtr)# manet peering selective redundancy 2`
		Enables selective peering for all MANET interfaces using this router process.
		Optional: Redundancy level configuration (valid range 0-10). Lower redundancy reduces OSPFv3 control-plane overhead. Higher levels increase control-plane redundancy.
		1—Default redundancy level (maintains two or more paths—one primary and one redundant path) for each one-hop OSPFv3 neighbor.
		The per-interface option adjusts the scope of peer selection to the interface level.
		By default, the peer-selection scope is per-area and across all MANET interfaces in a given area.

Step 6	`exit`
	Example:
	`Router(config-rtr)# exit`
		Exits global configuration mode and returns the router to privileged EXEC mode.

Preventing Full Peering over Poor Links

You can prevent full peering over poor links by performing this optional task, which configures the following:

- Configure OSPFv3 to wait for link metrics before considering a neighbor for OSPFv3 peering. (A cooperative radio may not advertise link metrics to the router before being discovered as a new OSPFv3 neighbor.)
- Configure OSPFv3 with a minimum metric threshold. If the radio-reported link metric is above this threshold, the neighbor will be held in 2-way state.

Note The commands in this task indicate IPv6. If you want to configure the OSPFv3 process for IPv4 instead, see the detailed steps for examples.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** [*type number*]
4. **ospfv3** [*process-id*] **manet peering link-metrics** [<threshold>]
5. **exit**

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 vmi1`
		Configures an interface type and enters interface configuration mode.

	Command or Action	Purpose
Step 4	**ospfv3** [*process-id*] **manet peering link-metrics** [<threshold>] **Example:** Router(config-if)# **ospfv3 6 manet peering link-metrics 200**	Requires receipt of link metrics from each radio before considering the new neighbor for selective peering. If the threshold (0-65535) is specified, the resultant link cost must be less than the threshold. Otherwise, the neighbor remains in a 2-way state.
Step 5	**exit** **Example:** Router(config-if)# **exit**	Exits global configuration mode and returns the router to privileged EXEC mode.

Fine-Tuning Selective Peering

This section describes how to optimize dynamic path costs by means of fine-tuning selective peering. Given a scenario without fine-tuning, each one-hop neighbor is awarded full-peering capabilities upon discovery, regardless of link cost:

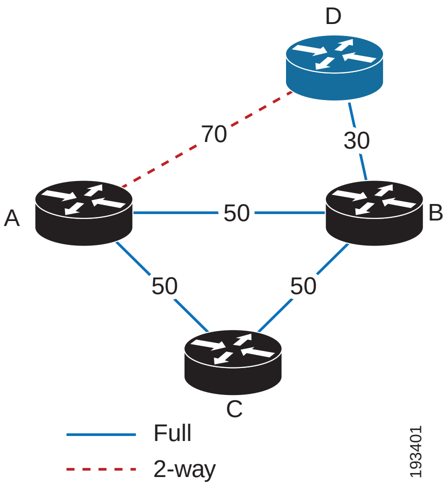
- Selective-peering redundancy level is greater than zero
- Link metrics are good (as determined by the configuration settings established in the "Preventing Full Peering over Poor Links" section on page 10-14)

As each additional neighbor is discovered, dynamic path costs are measurable immediately. To minimize path costs dynamically, you can configure the higher-cost links to remain in 2-way states until other peering opportunities become available.

Higher Costs without the Fine-Tuning

Consider the topology shown in Figure 10-1.

Figure 10-1 Peering Costs



Given the example shown in Figure 10-1, we have a static snapshot of a dynamic topology, beginning from this point:

- The redundancy level is set to 1 (the default value)—Router A attempts to maintain two paths for each one-hop neighbor.
- From the perspective of Router A, established neighbor sessions exist only with Routers B and C. Router D will join later.
- Router A has a full-peering relationship established with each of these known routers (B and C).
- The link cost for each of these neighbor sessions has a value of 50.
- At this point, only Router B has a link up to Router D—its peering relationship is full, and the link cost has a value of 30.

Change is then introduced between Router A and Router D:

1. Router D comes into radio range of Router A with a link cost of 70.
2. Router A establishes a full-peering relationship with this new neighbor. (The number of paths from Router A to Router D is currently 1 (through Router B).

The conclusion in this scenario (assigning full-peering capabilities between Routers A and D) is allowed given the original condition specified—the selective-peering redundancy level being greater than zero.

Improved Cost-Effectiveness through Fine-Tuning

To prevent the kind of scenario described in the "Higher Costs without the Fine-Tuning" section on page 10-15, you can fine-tune selective peering so that Routers A and D remain in a 2-way state until the link cost improves or an additional router comes into range—one with better link costs available to both routers (A and D).

Cost Thresholds for Redundant Paths

Setting a redundant-path cost threshold requires each redundant path to cost less than the existing, *best* path cost by a minimum value. For example, if the best link cost is 80, and you set the threshold value to 20, the new link cost must be less than 60 (80 minus 20).

Note The incremental improvement can be an absolute value or percentage.

Given the topology from Figure 10-1, if you set the redundant-path cost threshold to 20, you can prevent full peering between Routers A and D. This changes the outcome of our scenario, then, as follows:

1. Router D comes into radio range of Router A with a link cost of 70.
2. Selective peering compares link costs:
   - 80—Existing link cost between Routers A and D; the sum of link costs via Router B (50 + 30)
   - 70—The additional link cost between Routers A and D, if full peering is granted
3. The additional link cost (70) is incrementally better than the existing link cost (80) by a value of 10.
4. The incremental improvement (10) does not meet the minimum threshold (20); therefore, Routers A and D remain in the 2-way state.

Note The commands in this task indicate IPv6. If you want to configure the OSPFv3 process for IPv4 instead, see the detailed steps for examples.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** [*type number*]
4. **ospfv3** [*process-id*] **manet peering cost** {threshold <*0-65535*> | percent <*0-100*>}
5. **exit**

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 vmi1`
		Configures an interface type.

Step 4	`ospfv3 [process-id] manet peering cost {threshold <0-65535> | percent <0-100>}`
	Example:
	`Router(config-if)# ospfv3 6 manet cost percent 10`
		Requires redundant paths to have an incrementally better path cost than the current best path cost. The incremental improvement can be specified either as an absolute value (0-65535) or as a percentage (0-100) of the current best path cost.

Step 5	`exit`
	Example:
	`Router(config-if)# exit`
		Exits global configuration mode and returns the router to privileged EXEC mode.

Verifying OSPFv3 MANET Configuration and Operation

You can use any combination of the commands listed in this section to check the operation status of OSPFv3 MANET for IPv6 or IPv4. See Appendix A, "Command Reference" for detailed command reference.

Note You must be in privileged EXEC mode to enter the command listed in this section.

Command or Action	Purpose

`show run`
Example:
`Router#` **`show run`**
	Verify a configuration.

`show ospfv3` `[`*`process-id`*`]`
Example:
`Router#` **`show ospfv3 6`**
Example for IPv4:
`Router#` **`show ospfv3 4`**
	Displays general information about all OSPFv3 routing processes.

`show ospfv3 neighbor`
Example:
`Router#` **`show ospfv3 neighbor`**
	Displays OSPFv3 neighbor information per routing process.

`show ospfv3 neighbor detail`
Example:
`Router#` **`show ospfv3 neighbor detail`**
	Displays a detailed list of all neighbors.

`show ospfv3 neighbor manet`
Example:
`Router#` **`show ospfv3 neighbor manet`**
	Displays all neighbors in a MANET.

`show ospfv3` `[`*`process-id`*`]` **`interface`** `[interface-`*`type interface-number`*`]`
Example:
`Router#` **`show ospfv3 6 interface ethernet0/0`**
Example for IPv4:
`Router#` **`show ospfv3 4 interface ethernet0/0`**
	Displays information about OSPFv3 routing processes for an interface.

EXAMPLES

The examples in this section show how you can use the **show ospfv3** command to display general information about the OSPFv3 router process.

Example:

```
Router# show ospfv3
 Routing Process "ospfv3 6" with ID 1.1.1.1
Supports IPv6 Address Family
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 Initial SPF schedule delay 1000 msecs
 Minimum hold time between two consecutive SPFs 2000 msecs
 Maximum wait time between two consecutive SPFs 2000 msecs
 Minimum LSA interval 5 secs
 Minimum LSA arrival 1000 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 Number of external LSA 0. Checksum Sum 0x000000
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 Graceful restart helper support enabled
 Reference bandwidth unit is 100 mbps
 Relay willingness value is 128
 Pushback timer value is 2000 msecs
 Relay acknowledgement timer value is 1000 msecs
 LSA cache Disabled : current count 0, maximum 1000
 ACK cache Disabled : current count 0, maximum 1000
 Selective Peering is not enabled
 Hello requests and responses will be sent multicast
    Area BACKBONE(0) (Inactive)
        Number of interfaces in this area is 1
        SPF algorithm executed 0 times
        Number of LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

Example:

```
Router# show ospfv3 neighbor

            OSPFv3 Router with ID (1.1.1.1) (Process ID 6)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
2.2.2.2           0   FULL/  -        00:00:19    3               Ethernet0/0
```

Example:

```
Router# show ospfv3 neighbor manet

            OSPFv3 Router with ID (1.1.1.1) (Process ID 6)

Area BACKBONE(0) (Inactive)
Codes: D - cost dynamic default, R - received link cost,
       I - inherited from interface

Neighbor ID       State  Nbr Relay    Cost        Interface
 2.2.2.2           FULL      -      10   (I)    Ethernet0/0
```

Example:

```
Router# show ospfv3 interface e0/0
Ethernet0/0 is up, line protocol is up
  Link Local Address FE80::A8BB:CCFF:FE01:5500, Interface ID 3
  Area 0, Process ID 100, Instance ID 0, Router ID 1.1.1.1
  Network Type MANET, Cost: 10 (dynamic), Cost Hysteresis: Disabled
  Cost Weights: Throughput 100, Resources 100, Latency 100, L2-factor 100
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
    Hello due in 00:00:01
  Graceful restart helper support enabled
  Index 1/1/1, 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 2.2.2.2
  Suppress hello for 0 neighbor(s)
  Incremental Hello is enabled
  Local SCS number 1
  Relaying enabled
```