

# Brocade 5600 vRouter OSPFv3 Configuration Guide

Supporting Brocade 5600 vRouter 4.2R1

© 2016, Brocade Communications Systems, Inc. All Rights Reserved.

Brocade, Brocade Assurance, the B-wing symbol, ClearLink, DCX, Fabric OS, HyperEdge, ICX, MLX, MyBrocade, OpenScript, VCS, VDX, Vplane, and Vyatta are registered trademarks, and Fabric Vision is a trademark of Brocade Communications Systems, Inc., in the United States and/or in other countries. Other brands, products, or service names mentioned may be trademarks of others.

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. assume no liability or responsibility to any person or entity with respect to the accuracy of this document or any loss, cost, liability, or damages arising from the information contained herein or the computer programs that accompany it.

The product described by this document may contain open source software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit <http://www.brocade.com/support/oscd>.

# Contents

---

<b>Preface</b> .....	<b>7</b>
Document conventions.....	7
Text formatting conventions.....	7
Command syntax conventions.....	7
Notes, cautions, and warnings.....	8
Brocade resources.....	8
Contacting Brocade Technical Support.....	8
Brocade customers.....	8
Brocade OEM customers.....	9
Document feedback.....	9
<b>About This Guide</b> .....	<b>11</b>
<b>OSPFv3 Configuration</b> .....	<b>13</b>
OSPFv3 overview.....	13
Support for OSPFv3 address families.....	14
Supported standard.....	14
Configuring OSPFv3.....	14
Basic OSPFv3 configuration.....	15
Verifying the OSPFv3 configuration.....	18
<b>OSPFv3 Commands</b> .....	<b>21</b>
monitor protocol ospfv3 background.....	23
monitor protocol ospfv3 disable events.....	24
monitor protocol ospfv3 disable ifsm.....	25
monitor protocol ospfv3 disable lsa.....	26
monitor protocol ospfv3 disable nsm.....	27
monitor protocol ospfv3 disable nsm.....	28
monitor protocol ospfv3 disable packet all.....	29
monitor protocol ospfv3 disable packet dd.....	30
monitor protocol ospfv3 disable packet detail.....	31
monitor protocol ospfv3 disable packet hello.....	32
monitor protocol ospfv3 disable packet ls-ack.....	33
monitor protocol ospfv3 disable packet ls-request.....	34
monitor protocol ospfv3 disable packet ls-update.....	35
monitor protocol ospfv3 disable packet recv.....	36
monitor protocol ospfv3 disable packet send.....	37
monitor protocol ospfv3 disable route.....	38
monitor protocol ospfv3 enable events.....	39
monitor protocol ospfv3 enable ifsm.....	40
monitor protocol ospfv3 enable lsa.....	41
monitor protocol ospfv3 enable nsm.....	42
monitor protocol ospfv3 enable nsm.....	43
monitor protocol ospfv3 enable packet all.....	44
monitor protocol ospfv3 enable packet dd.....	45
monitor protocol ospfv3 enable packet detail.....	46
monitor protocol ospfv3 enable packet hello.....	47
monitor protocol ospfv3 enable packet ls-ack.....	48

monitor protocol ospfv3 enable packet ls-request.....	49
monitor protocol ospfv3 enable packet ls-update.....	50
monitor protocol ospfv3 enable packet recv.....	51
monitor protocol ospfv3 enable packet send.....	52
monitor protocol ospfv3 enable route.....	53
protocols ospfv3 abr-type.....	54
protocols ospfv3 auto-cost reference-bandwidth <bandwidth>.....	55
protocols ospfv3 default-information originate.....	56
protocols ospfv3 default-metric <metric>.....	58
protocols ospfv3 distance.....	59
protocols ospfv3 distribute-list <name>.....	61
protocols ospfv3 log.....	62
protocols ospfv3 log events.....	64
protocols ospfv3 log ifsm.....	66
protocols ospfv3 log lsa.....	67
protocols ospfv3 log nasm.....	69
protocols ospfv3 log nsm.....	70
protocols ospfv3 log packet.....	71
protocols ospfv3 log rib.....	73
protocols ospfv3 log route.....	74
protocols ospfv3 passive-interface.....	76
protocols ospfv3 passive-interface-exclude.....	77
protocols ospfv3 process <process-id> address-family ipv4 unicast.....	79
protocols ospfv3 process <process-id> address-family ipv4 unicast area <area-id> range.....	81
protocols ospfv3 process <process-id> address-family ipv4 unicast default-information originate.....	83
protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute.....	85
protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute bgp.....	87
protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute connected.....	89
protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute kernel.....	91
protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute rip.....	93
protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute static.....	95
protocols ospfv3 process <process-id> address-family ipv4 unicast summary-address.....	97
protocols ospfv3 process <process-id> passive-interface-exclude.....	99
protocols ospfv3 redistribute bgp.....	101
protocols ospfv3 redistribute connected.....	103
protocols ospfv3 redistribute kernel.....	105
protocols ospfv3 redistribute rip.....	107
protocols ospfv3 redistribute static.....	109
protocols ospfv3 router-id <router-id>.....	111
protocols ospfv3 summary-address <address> <action>.....	113
show ipv6 ospfv3.....	115
show ipv6 ospfv3 database.....	116
show ipv6 ospfv3 database adv-router.....	118
show ipv6 ospfv3 database external.....	119
show ipv6 ospfv3 database inter-prefix.....	120
show ipv6 ospfv3 database inter-router.....	121
show ipv6 ospfv3 database intra-prefix.....	122
show ipv6 ospfv3 database link.....	123
show ipv6 ospfv3 database max-age.....	124
show ipv6 ospfv3 database network.....	125

show ipv6 ospfv3 database nssa-external.....	126
show ipv6 ospfv3 database router.....	127
show ipv6 ospfv3 database self-originate.....	128
show ipv6 ospfv3 interface.....	129
show ipv6 ospfv3 neighbor.....	130
show ipv6 ospfv3 route.....	131
show ipv6 ospfv3 topology.....	132
show ipv6 ospfv3 virtual-links.....	133
show ipv6 route ospfv3.....	134
show monitoring protocols ospfv3.....	135
<b>OSPFv3 Area Commands.....</b>	<b>137</b>
protocols ospfv3 area <area-id>.....	138
protocols ospfv3 area <area-id> nssa.....	139
protocols ospfv3 area <area-id> nssa default-cost.....	140
protocols ospfv3 area <area-id> nssa default-information-originate.....	141
protocols ospfv3 area <area-id> nssa default-information-originate metric.....	142
protocols ospfv3 area <area-id> nssa default-information-originate metric-type.....	143
protocols ospfv3 area <area-id> nssa no-redistribution.....	144
protocols ospfv3 area <area-id> nssa no-summary.....	145
protocols ospfv3 area <area-id> nssa stability-interval.....	146
protocols ospfv3 area <area-id> nssa translator-role.....	147
protocols ospfv3 area <area-id> range.....	148
protocols ospfv3 area <area-id> stub.....	150
protocols ospfv3 area <area-id> virtual-link <router-id>.....	152
protocols ospfv3 area <area-id> virtual-link <router-id> dead-interval <interval>.....	153
protocols ospfv3 area <area-id> virtual-link <router-id> hello-interval <interval>.....	155
protocols ospfv3 area <area-id> virtual-link <router-id> retransmit-interval <interval>.....	157
protocols ospfv3 area <area-id> virtual-link <router-id> transmit-delay <delay>.....	159
<b>OSPFv3 Interface Commands.....</b>	<b>161</b>
interfaces <interface> ipv6 ospfv3.....	162
interfaces <interface> ipv6 ospfv3 cost <cost>.....	163
interfaces <interface> ipv6 ospfv3 dead-interval <interval>.....	165
interfaces <interface> ipv6 ospfv3 hello-interval <interval>.....	167
interfaces <interface> ipv6 ospfv3 link-lsa-suppression enable.....	169
interfaces <interface> ipv6 ospfv3 mtu-ignore.....	170
interfaces <interface> ipv6 ospfv3 neighbor.....	172
interfaces <interface> ipv6 ospfv3 network <type>.....	173
interfaces <interface> ipv6 ospfv3 priority <priority>.....	175
interfaces <interface> ipv6 ospfv3 retransmit-interval <interval>.....	177
interfaces <interface> ipv6 ospfv3 transmit-delay <delay>.....	179
<b>Supported Interface Types.....</b>	<b>181</b>
<b>List of Acronyms.....</b>	<b>183</b>



# Preface

---

- Document conventions..... 7
- Brocade resources..... 8
- Contacting Brocade Technical Support..... 8
- Document feedback..... 9

## Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Brocade technical documentation.

## Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used in the flow of the text to highlight specific words or phrases.

Format	Description
<b>bold text</b>	Identifies command names Identifies keywords and operands Identifies the names of user-manipulated GUI elements
<i>italic text</i>	Identifies text to enter at the GUI Identifies emphasis Identifies variables
Courier font	Identifies document titles Identifies CLI output Identifies command syntax examples

## Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
<b>bold text</b>	Identifies command names, keywords, and command options.
<i>italic text</i>	Identifies a variable.
value	In Fibre Channel products, a fixed value provided as input to a command option is printed in plain text, for example, <b>--show</b> WWN.
[ ]	Syntax components displayed within square brackets are optional.
{ x   y   z }	Default responses to system prompts are enclosed in square brackets. A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x   y	In Fibre Channel products, square brackets may be used instead for this purpose. A vertical bar separates mutually exclusive elements.
< >	Nonprinting characters, for example, passwords, are enclosed in angle brackets.

Convention	Description
...	Repeat the previous element, for example, <i>member{member...}</i> .
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

## Notes, cautions, and warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

### NOTE

A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

### ATTENTION

An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.



### CAUTION

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.



### DANGER

*A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.*

## Brocade resources

Visit the Brocade website to locate related documentation for your product and additional Brocade resources.

You can download additional publications supporting your product at [www.brocade.com](http://www.brocade.com). Select the Brocade Products tab to locate your product, then click the Brocade product name or image to open the individual product page. The user manuals are available in the resources module at the bottom of the page under the Documentation category.

To get up-to-the-minute information on Brocade products and resources, go to [MyBrocade](http://MyBrocade). You can register at no cost to obtain a user ID and password.

Release notes are available on [MyBrocade](http://MyBrocade) under Product Downloads.

White papers, online demonstrations, and data sheets are available through the [Brocade website](http://Brocade website).

## Contacting Brocade Technical Support

As a Brocade customer, you can contact Brocade Technical Support 24x7 online, by telephone, or by e-mail. Brocade OEM customers contact their OEM/Solutions provider.

## Brocade customers

For product support information and the latest information on contacting the Technical Assistance Center, go to <http://www.brocade.com/services-support/index.html>.

If you have purchased Brocade product support directly from Brocade, use one of the following methods to contact the Brocade Technical Assistance Center 24x7.



Online	Telephone	E-mail
Preferred method of contact for non-urgent issues: <ul style="list-style-type: none"> <li>• <a href="#">My Cases</a> through MyBrocade</li> <li>• <a href="#">Software downloads</a> and licensing tools</li> <li>• <a href="#">Knowledge Base</a></li> </ul>	Required for Sev 1-Critical and Sev 2-High issues: <ul style="list-style-type: none"> <li>• Continental US: 1-800-752-8061</li> <li>• Europe, Middle East, Africa, and Asia Pacific: +800-AT FIBREE (+800 28 34 27 33)</li> <li>• For areas unable to access toll free number: +1-408-333-6061</li> <li>• <a href="#">Toll-free numbers</a> are available in many countries.</li> </ul>	<a href="mailto:support@brocade.com">support@brocade.com</a> Please include: <ul style="list-style-type: none"> <li>• Problem summary</li> <li>• Serial number</li> <li>• Installation details</li> <li>• Environment description</li> </ul>

## Brocade OEM customers

If you have purchased Brocade product support from a Brocade OEM/Solution Provider, contact your OEM/Solution Provider for all of your product support needs.

- OEM/Solution Providers are trained and certified by Brocade to support Brocade® products.
- Brocade provides backline support for issues that cannot be resolved by the OEM/Solution Provider.
- Brocade Supplemental Support augments your existing OEM support contract, providing direct access to Brocade expertise. For more information, contact Brocade or your OEM.
- For questions regarding service levels and response times, contact your OEM/Solution Provider.

## Document feedback

To send feedback and report errors in the documentation you can use the feedback form posted with the document or you can e-mail the documentation team.

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. You can provide feedback in two ways:

- Through the online feedback form in the HTML documents posted on [www.brocade.com](http://www.brocade.com).
- By sending your feedback to [documentation@brocade.com](mailto:documentation@brocade.com).

Provide the publication title, part number, and as much detail as possible, including the topic heading and page number if applicable, as well as your suggestions for improvement.



# About This Guide

---

This guide describes how to configure OSPFv3 on the Brocade vRouter (referred to as a virtual router, vRouter, or router in the guide).



# OSPFv3 Configuration

---

• <a href="#">OSPFv3 overview</a> .....	13
• <a href="#">Support for OSPFv3 address families</a> .....	14
• <a href="#">Supported standard</a> .....	14
• <a href="#">Configuring OSPFv3</a> .....	14

## OSPFv3 overview

Open Shortest Path First Version 3 (OSPFv3) is a link-state Interior Gateway Protocol (IGP) developed by IETF and is based on OSPF Version 2 (OSPFv2), which supports IPv4. For more information about OSPFv2, refer to *Brocade 5600 vRouter OSPF Configuration Guide*. OSPFv3 adds support for IPv6 in the OSPF routing protocol and is described in RFC 5340. OSPFv3 works as in OSPFv2, that is, OSPFv3 router IDs, area IDs, and LSA link-state IDs are 32 bits as in OSPFv2 IPv4. In addition, all optional capabilities of OSPFv2 for IPv4, for example, NSSA, are supported in OSPFv3 for IPv6.

However, some significant changes exist in OSPFv3 for IPv6 and these changes include the following:

- Router LSAs and network LSAs no longer carry prefix information. In OSPFv3, these LSAs carry only the topology information.
- New and modified LSAs have been created to handle the flow of IPv6 addresses and prefixes in an OSPFv3 network.
- OSPFv3 runs on a per-link basis, instead of a per-IP-subnet basis. Two nodes can communicate over a link even when they do not share a common IPv6 prefix.
- Multiple OSPFv3 instances can run over the same link.
- OSPFv3 supports stub areas.
- IPv6 link-local addresses are used for OSPFv3 neighbor exchanges, except over virtual links. Link-local addresses have local significance only.
- The 32-bit router ID in OSPFv3 always identifies neighboring routers.
- The flooding scope for LSAs is generalized into three categories for OSPFv3:
  - Link-local scope. The OSPFv3 packet is flooded only to the members connected on a link.
  - Area scope. The OSPFv3 packet is flooded to all members of an OSPFv3 area.
  - AS scope. The OSPFv3 packet is flooded to all members of an AS.
- Authentication is removed from the OSPFv3 protocol. Instead, OSPFv3 relies on the AH and ESP portions of the IPsec for all authentication tasks in IPv6.
- LSPs and traffic engineering are not supported in OSPFv3.
- In OSPFv3, the Options field is available in only Hello packets, DD packets, Router LSAs, Network LSAs, Inter-Area-Router LSAs, and Link LSAs.
- OSPFv3 has two new LSAs: Link LSA and Intra Area Prefix LSA.
- OSPFv3 communicates over IPv6, but can carry routing data for other address families, most notably IPv4.

OSPFv3 adds support for address families which helps OSPFv3 IPv6 networks to support both IPv6 and IPv4 nodes. For more information on this feature, refer to [Support for OSPFv3 address families](#) on page 14.

## Support for OSPFv3 address families

OSPFv3 communicates with peer routers over IPv6, but can carry routing data for other address families. By default, OSPFv3 supports the IPv6 unicast address family. However, OSPFv3 can support other address families when configured to do so. The Brocade 5600 vRouter OSPFv3 implementation provides support for carrying IPv4 and IPv6 unicast routing data.

It is important to understand that instance identifiers not only identify a routing instance, but also identify an address family. The following are the instance IDs breakdown:

- Instance ID 0 to 31: IPv6 unicast AF
- Instance ID 32 to 63: IPv6 multicast AF
- Instance ID 64 to 95: IPv4 unicast AF
- Instance ID 96 to 127: IPv4 multicast AF

OSPFv3 supports multiple OSPFv3 processes and each process may support one address family. Multiple routing instances within each OSPFv3 process are also supported.

## Supported standard

The Brocade Communications Systems, Inc. implementation of OSPFv3 complies with the following standards:

- RFC 5340: OSPF for IPv6
- RFC 5838: Support of Address Families in OSPFv3

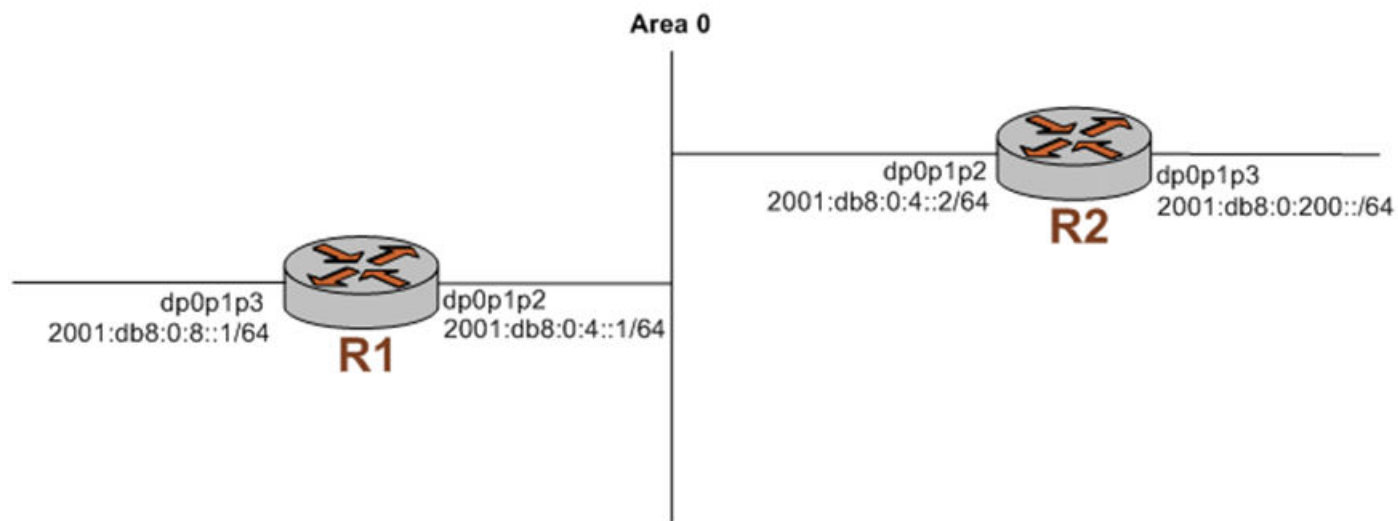
## Configuring OSPFv3

This section provides the following topics:

- [Basic OSPFv3 configuration](#) on page 15
- [Verifying the OSPFv3 configuration](#) on page 18

This section presents a sample configuration for OSPFv3. The configuration example is based on the following reference diagram.

FIGURE 1 OSPFv3 configuration reference diagram



## Basic OSPFv3 configuration

In this section, you configure OSPFv3 on the routers labeled R1 and R2 in the reference network diagram.

To create a basic OSPFv3 configuration, perform the following steps in configuration mode:

Router	Step	Commands
Configure Interface IP addresses and OSPFv3 Configuration		
R1	Set an IPv6 address on interface dp0p1p2.	<pre>vyatta@R1# set interfaces dataplane dp0p1p2 address 2001:db8:0:4::1/64</pre>
R1	Set an IPv6 address on interface dp0p1p3.	<pre>vyatta@R1# set interfaces dataplane dp0p1p3 address 2001:db8:0:8::1/64</pre>
R1	Set the OSPFv3 area on interface dp0p1p2.	<pre>vyatta@R1# set interfaces dataplane dp0p1p2 ipv6 ospfv3 area 0.0.0.0</pre>
R1	Set the OSPFv3 area on interface dp0p1p3.	<pre>vyatta@R1# set interfaces dataplane dp0p1p3 ipv6 ospfv3 area 0.0.0.0</pre>
R1	Set the OSPFv3 area for a given process ID and instance ID on interface dp0p1p2.	<pre>vyatta@R1# set interfaces dataplane dp0p1p2 ipv6 ospfv3 process labR instance-id 3 area 0.0.0.0</pre>
R1	Set the OSPFv3 area for a given process ID and instance ID on interface dp0p1p3.	<pre>vyatta@R1# set interfaces dataplane dp0p1p3 ipv6 ospfv3 process labR instance-id 3 area 0.0.0.0</pre>

R1	Display the configuration.	<pre> vyatta@R1# show interfaces dataplane dp0p1p2 {   address 2001:db8:0:4::1/64   duplex auto   hw-id 00:0c:29:09:3e:73   ipv6 {     dup-addr-detect-     transmits 1     ospfv3 {       area 0.0.0.0       process labR {         instance-id 3 {           area           0.0.0.0         }       }     }   }   dataplane dp0p1p3 {     address 2001:db8:0:8::1/64     duplex auto     hw-id 00:0c:29:09:3e:7d     ipv6 {       dup-addr-detect-       transmits 1       ospfv3 {         area 0.0.0.0         process labR {           instance-id 3 {             area             0.0.0.0           }         }       }     }   } } </pre>
Configure OSPFv3 Protocol Commands		
R1	Specify the OSPFv3 address range and advertise this range.	<pre> vyatta@R1# set protocols ospfv3 process labR area 0.0.0.0 range 2001::1/128 advertise </pre>
R1	Set the global router ID for OSPFv3.	<pre> vyatta@R1# set protocols ospfv3 process labR router-id 10.10.10.10 </pre>
R1	Commit the configuration.	<pre> vyatta@R1# commit </pre>
R1	Display the configuration.	<pre> vyatta@R1# show protocols ospfv3 {   process labR {     area 0.0.0.0 {     }   }   router-id 10.10.10.10 } </pre>
Configure Interface IP Addresses and OSPFv3 Configuration		
R2	Set an IPv6 address on interface dp0p1p2.	<pre> vyatta@R2# set interfaces dataplane dp0p1p2 address 2001:db8:0:4::2/64 </pre>



R2	Set an IPv6 address on interface dp0p1p3.	<pre>vyatta@R2# set interfaces dataplane dp0p1p3 address 2001:db8:0:200::/64</pre>
R2	Set the OSPFv3 area on interface dp0p1p2.	<pre>vyatta@R2# set interfaces dataplane dp0p1p2 ipv6 ospfv3 area 0.0.0.0</pre>
R2	Set the OSPFv3 area for a given process ID and instance ID on interface dp0p1p2.	<pre>vyatta@R2# set interfaces dataplane dp0p1p2 ipv6 ospfv3 process labR instance-id 3 area 0.0.0.0</pre>
R2	Display the interface configuration.	<pre>vyatta@R2# show interfaces dataplane dp0p1p2 { address 2001:db8:0:4::2/64 duplex auto hw-id 00:0c:29:4b:d8:ff ipv6 { dup-addr-detect- transmits 1 ospfv3 { area 0.0.0.0 process labR { instance-id 3 { area 0.0.0.0 } } } } } dataplane dp0p1p3 { address 2001:db8:0:200::/64 duplex auto hw-id 00:0c:29:4b:d8:09 speed auto }</pre>
Configure OSPFv3 Protocol Commands		
R2	Specify the OSPFv3 address range and advertise this range.	<pre>vyatta@R2# set protocols ospfv3 process labR area 0.0.0.0 range 2001::2/128 advertise</pre>
R2	Set the global OSPFv3 parameter of a router ID.	<pre>vyatta@R2# set protocols ospfv3 process labR router-id 10.10.10.11</pre>
R2	Commit the configuration.	<pre>vyatta@R2# commit</pre>
R2	Display the configuration.	<pre>vyatta@R2# show protocols ospfv3 { process labR { area 0.0.0.0 { range 2001::2/128 { advertise } } router-id 10.10.10.11 } }</pre>

## Verifying the OSPFv3 configuration

The following operational mode commands can be used to verify the OSPFv3 configuration.

### R1: show ipv6 ospfv3 neighbor

The following example shows the output of the **show ipv6 ospfv3 neighbor** command for router R1.

#### Verifying OSPFv3 on R1: "show ipv6 ospfv3 neighbor"

```
vyatta@R1:~$ show ipv6 ospfv3 neighbor
OSPFv3 Process (labR)
Neighbor ID      Pri   State           Dead Time   Interface   Instance ID
10.10.10.11     1     Full/DR         00:00:34   dp0p1p2 3
```

### R1: show ipv6 route

The following example shows the output of the **show ipv6 route** command for router R1.

#### Verifying OSPFv3 on R1: "show ipv6 route"

```
vyatta@R1:~$ show ipv6 route
IPv6 Routing Table
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
      IA - OSPF inter area, E1 - OSPF external type 1,
      E2 - OSPF external type 2, I - IS-IS, B - BGP
      > - selected route, * - FIB route, p - stale info
Timers: Uptime
C   *> ::1/128 is directly connected, lo
O   2001:db8:0:4::/64 [110/1] is directly connected, dp0p1p2, 23:39:41
C   *> 2001:db8:0:4::/64 is directly connected, dp0p1p2
O   2001:db8:0:8::/64 [110/1] is directly connected, dp0p1p3, 23:39:41
C   *> 2001:db8:0:8::/64 is directly connected, dp0p1p3
C   * fe80::/64 is directly connected, dp0p1p2
C   * fe80::/64 is directly connected, dp0p1p1
C   *> fe80::/64 is directly connected, dp0p1p3
K   *> ff00::/8 is directly connected, dp0p1p2
K   * fe00::/8 is directly connected, dp0p1p1
K   * ff00::/8 is directly connected, dp0p1p3
```

### R2: show ipv6 ospfv3 neighbor

The following example shows the output of the **show ipv6 ospfv3 neighbor** command for router R2.

#### Verifying OSPFv3 on R2: "show ipv6 ospfv3 neighbor"

```
vyatta@R2:~$ show ipv6 ospfv3 neighbor
OSPFv3 Process (labR)
Neighbor ID      Pri   State           Dead Time   Interface   Instance ID
10.10.10.10     1     Full/Backup     00:00:36   dp0p1p2 3
```

### R2: show ipv6 route

The following example shows the output of the **show ipv6 route** command for router R2.

#### Verifying OSPFv3 on R2: "show ipv6 route"

```
vyatta@R2:~$ show ipv6 route
IPv6 Routing Table
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
      IA - OSPF inter area, E1 - OSPF external type 1,
      E2 - OSPF external type 2, I - IS-IS, B - BGP
      > - selected route, * - FIB route, p - stale info
```

```

Timers: Uptime
C  *> ::1/128 is directly connected, lo
O  2001:db8:0:4::/64 [110/1] is directly connected, dp0p1p2, 23:44:53
C  *> 2001:db8:0:4::/64 is directly connected, dp0p1p2
O  *> 2001:db8:0:8::/64 [110/2] via fe80::20c:29ff:fe09:3e73, dp0p1p2, 23:44:02
C  *> 2001:db8:0:200::/64 is directly connected, dp0p1p3
C  *> 2004::/64 is directly connected, lo
C  * fe80::/64 is directly connected, dp0p1p2
C  * fe80::/64 is directly connected, dp0p1p1
C  *> fe80::/64 is directly connected, dp0p1p3
K  *> ff00::/8 is directly connected, dp0p1p2
K  * ff00::/8 is directly connected, dp0p1p1
K  * ff00::/8 is directly connected, dp0p1p3

```

## R2: ping 2001:db8:0:8::1

Using the **ping** command from R2, we can verify that we can reach hosts on remote networks. In this case we ping an IP address on R1. This verification is shown in the following example.

### Verifying OSPF on R2: "ping6 2001:db8:0:8::1"

```

vyatta@R2# ping6 2001:db8:0:8::1
PING 2001:db8:0:8::1(2001:db8:0:8::1) 56 data bytes
64 bytes from 2001:db8:0:8::1: icmp_seq=1 ttl=64 time=0.460 ms
64 bytes from 2001:db8:0:8::1: icmp_seq=2 ttl=64 time=0.153 ms
64 bytes from 2001:db8:0:8::1: icmp_seq=3 ttl=64 time=0.209 ms
64 bytes from 2001:db8:0:8::1: icmp_seq=4 ttl=64 time=0.156 ms
64 bytes from 2001:db8:0:8::1: icmp_seq=5 ttl=64 time=0.163 ms
64 bytes from 2001:db8:0:8::1: icmp_seq=6 ttl=64 time=0.159 ms
--- 2001:db8:0:8::1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 4997ms
rtt min/avg/max/mdev = 0.153/0.216/0.460/0.111 ms

```

The output confirms that the OSPFv3 configuration is working and that we are able to reach a remote network.



# OSPFv3 Commands

---

• monitor protocol ospfv3 background.....	23
• monitor protocol ospfv3 disable events.....	24
• monitor protocol ospfv3 disable ifsm.....	25
• monitor protocol ospfv3 disable lsa.....	26
• monitor protocol ospfv3 disable nsm.....	27
• monitor protocol ospfv3 disable nsm.....	28
• monitor protocol ospfv3 disable packet all.....	29
• monitor protocol ospfv3 disable packet dd.....	30
• monitor protocol ospfv3 disable packet detail.....	31
• monitor protocol ospfv3 disable packet hello.....	32
• monitor protocol ospfv3 disable packet ls-ack.....	33
• monitor protocol ospfv3 disable packet ls-request.....	34
• monitor protocol ospfv3 disable packet ls-update.....	35
• monitor protocol ospfv3 disable packet recv.....	36
• monitor protocol ospfv3 disable packet send.....	37
• monitor protocol ospfv3 disable route.....	38
• monitor protocol ospfv3 enable events.....	39
• monitor protocol ospfv3 enable ifsm.....	40
• monitor protocol ospfv3 enable lsa.....	41
• monitor protocol ospfv3 enable nsm.....	42
• monitor protocol ospfv3 enable nsm.....	43
• monitor protocol ospfv3 enable packet all.....	44
• monitor protocol ospfv3 enable packet dd.....	45
• monitor protocol ospfv3 enable packet detail.....	46
• monitor protocol ospfv3 enable packet hello.....	47
• monitor protocol ospfv3 enable packet ls-ack.....	48
• monitor protocol ospfv3 enable packet ls-request.....	49
• monitor protocol ospfv3 enable packet ls-update.....	50
• monitor protocol ospfv3 enable packet recv.....	51
• monitor protocol ospfv3 enable packet send.....	52
• monitor protocol ospfv3 enable route.....	53
• protocols ospfv3 abr-type.....	54
• protocols ospfv3 auto-cost reference-bandwidth <bandwidth>.....	55
• protocols ospfv3 default-information originate.....	56
• protocols ospfv3 default-metric <metric>.....	58
• protocols ospfv3 distance.....	59
• protocols ospfv3 distribute-list <name>.....	61
• protocols ospfv3 log.....	62
• protocols ospfv3 log events.....	64
• protocols ospfv3 log ifsm.....	66
• protocols ospfv3 log lsa.....	67
• protocols ospfv3 log nsm.....	69
• protocols ospfv3 log nsm.....	70
• protocols ospfv3 log packet.....	71
• protocols ospfv3 log rib.....	73
• protocols ospfv3 log route.....	74
• protocols ospfv3 passive-interface.....	76
• protocols ospfv3 passive-interface-exclude.....	77
• protocols ospfv3 process <process-id> address-family ipv4 unicast.....	79

• protocols ospfv3 process <process-id> address-family ipv4 unicast area <area-id> range.....	81
• protocols ospfv3 process <process-id> address-family ipv4 unicast default-information originate.....	83
• protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute.....	85
• protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute bgp.....	87
• protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute connected.....	89
• protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute kernel.....	91
• protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute rip.....	93
• protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute static.....	95
• protocols ospfv3 process <process-id> address-family ipv4 unicast summary-address.....	97
• protocols ospfv3 process <process-id> passive-interface-exclude.....	99
• protocols ospfv3 redistribute bgp.....	101
• protocols ospfv3 redistribute connected.....	103
• protocols ospfv3 redistribute kernel.....	105
• protocols ospfv3 redistribute rip.....	107
• protocols ospfv3 redistribute static.....	109
• protocols ospfv3 router-id <router-id>.....	111
• protocols ospfv3 summary-address <address> <action>.....	113
• show ipv6 ospfv3.....	115
• show ipv6 ospfv3 database.....	116
• show ipv6 ospfv3 database adv-router.....	118
• show ipv6 ospfv3 database external.....	119
• show ipv6 ospfv3 database inter-prefix.....	120
• show ipv6 ospfv3 database inter-router.....	121
• show ipv6 ospfv3 database intra-prefix.....	122
• show ipv6 ospfv3 database link.....	123
• show ipv6 ospfv3 database max-age.....	124
• show ipv6 ospfv3 database network.....	125
• show ipv6 ospfv3 database nssa-external.....	126
• show ipv6 ospfv3 database router.....	127
• show ipv6 ospfv3 database self-originate.....	128
• show ipv6 ospfv3 interface.....	129
• show ipv6 ospfv3 neighbor.....	130
• show ipv6 ospfv3 route.....	131
• show ipv6 ospfv3 topology.....	132
• show ipv6 ospfv3 virtual-links.....	133
• show ipv6 route ospfv3.....	134
• show monitoring protocols ospfv3.....	135

## monitor protocol ospfv3 background

Starts and stops background monitoring of OSPFv3 packets.

### Syntax

```
monitor protocol ospfv3 background { start | stop }
```

### Parameters

**start**

Starts background monitoring of OSPFv3 packets.

**stop**

Stops background monitoring of OSPFv3 packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to start and stop background monitoring of OSPFv3 packets.

## monitor protocol ospfv3 disable events

Disables the generation of debug messages related to OSPFv3 events.

### Syntax

```
monitor protocol ospfv3 disable events [ abr | asbr | nssa | os | router | vlink ]
```

### Command Default

OSPFv3 event debugging is disabled.

### Parameters

**abr**

Disables debugging of Area Border Router (ABR) events.

**asbr**

Disables debugging of Autonomous System Boundary Router (ASBR) events.

**nssa**

Disables debugging of not-so-stubby area (NSSA) events.

**os**

Disables debugging of OS-interaction events.

**router**

Disables debugging of other router events.

**vlink**

Disables debugging of virtual-link events.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 events.



## monitor protocol ospfv3 disable ifsm

Disables the generation of debug messages related to OSPFv3 Interface Finite State Machine (IFSM) events.

### Syntax

```
monitor protocol ospfv3 disable ifsm [ events | status | timers ]
```

### Command Default

OSPFv3 IFSM event debugging is disabled.

### Parameters

**events**

Disables debugging of IFSM events.

**status**

Disables debugging of IFSM status information.

**timers**

Disables debugging of IFSM timer information.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 IFSM events.

## monitor protocol ospfv3 disable lsa

Disables the generation of debug messages related to OSPFv3 Link State Advertisements (LSAs).

### Syntax

```
monitor protocol ospfv3 disable lsa [ flooding | generate | install | maxage | refresh ]
```

### Command Default

OSPFv3 LSA event debugging is disabled.

### Parameters

**flooding**

Disables debugging of OSPFv3 LSA flood events.

**generate**

Disables debugging of OSPFv3 LSA generation.

**install**

Disables debugging of OSPFv3 LSA installation.

**maxage**

Disables debugging of maximum age processing related OSPFv3 LSA.

**refresh**

Disables debugging of OSPFv3 LSA refreshes.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 LSAs.

## monitor protocol ospfv3 disable nfsm

Disables the generation of debug messages related to OSPFv3 Neighbor Finite State Machines (NFSMs).

### Syntax

```
monitor protocol ospfv3 disable nfsm [ events | status | timers ]
```

### Command Default

OSPFv3 NFSM debugging is disabled.

### Parameters

**events**

Disables debugging of OSPFv3 NFSM events.

**status**

Disables debugging of OSPFv3 NFSM status.

**timers**

Disables debugging of OSPFv3 NFSM timers.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 NFSM.

## monitor protocol ospfv3 disable nsm

Disables the generation of debug messages related to the OSPFv3 Network Services Module (NSM).

### Syntax

```
monitor protocol ospfv3 disable nsm [ interface | redistribute ]
```

### Command Default

OSPFv3 NSM debugging is disabled.

### Parameters

#### **interface**

Disables debugging of OSPFv3 NSM interface events.

#### **redistribute**

Disables debugging of OSPFv3 NSM redistribute events.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to the OSPFv3 NSM.

## monitor protocol ospfv3 disable packet all

Disables the generation of debug messages related to all OSPFv3 packets that are arriving at and leaving a router.

### Syntax

```
monitor protocol ospfv3 disable packet all [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are disabled for all OSPFv3 packets at a medium level of detail.

### Parameters

**detail**

Disables detailed debugging of all OSPFv3 packets, both sent and received.

**recv**

Disables debugging of all received OSPFv3 packets.

**detail**

Disables detailed debugging of all received OSPFv3 packets.

**send**

Disables debugging of all transmitted OSPFv3 packets.

**detail**

Disables detailed debugging of all transmitted OSPFv3 packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to all OSPFv3 packets that are arriving at and leaving a router.

## monitor protocol ospfv3 disable packet dd

Disables the generation of debug messages related to OSPFv3 Database Description (DD) packets.

### Syntax

```
monitor protocol ospfv3 disable packet dd [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are disabled for OSPFv3 DD packets at a medium level of detail.

### Parameters

**detail**

Disables detailed debugging of OSPFv3 DD packets, both sent and received.

**recv**

Disables debugging of received OSPFv3 DD packets.

**detail**

Disables detailed debugging of received OSPFv3 DD packets.

**send**

Disables debugging of transmitted OSPFv3 DD packets.

**detail**

Disables detailed debugging of transmitted OSPFv3 DD packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 DD packets.

## monitor protocol ospfv3 disable packet detail

Disables the generation of debug messages related to OSPFv3 detailed packets.

### Syntax

```
monitor protocol ospfv3 disable packet detail
```

### Command Default

Debug messages are disabled for OSPFv3 detailed packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 detailed packets.

## monitor protocol ospfv3 disable packet hello

Disables the generation of debug messages related to OSPFv3 Hello packets.

### Syntax

```
monitor protocol ospfv3 disable packet hello [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are disabled for OSPFv3 Hello packets at a medium level of detail.

### Parameters

**detail**

Disables detailed debugging of OSPFv3 Hello packets, both sent and received.

**recv**

Disables debugging of received OSPFv3 Hello packets.

**detail**

Disables detailed debugging of received OSPFv3 Hello packets.

**send**

Disables debugging of transmitted OSPFv3 Hello packets.

**detail**

Disables detailed debugging of transmitted OSPFv3 Hello packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 Hello packets.



## monitor protocol ospfv3 disable packet ls-ack

Disables the generation of debug messages related to OSPFv3 link-state acknowledgement (LS Ack) packets.

### Syntax

```
monitor protocol ospfv3 disable packet ls-ack [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are disabled for OSPFv3 LS Ack packets at a medium level of detail.

### Parameters

**detail**

Disables detailed debugging of OSPFv3 LS Ack packets, both sent and received.

**recv**

Disables debugging of received OSPFv3 LS Ack packets.

**detail**

Disables detailed debugging of received OSPFv3 LS Ack packets.

**send**

Disables debugging of transmitted OSPFv3 LS Ack packets.

**detail**

Disables detailed debugging of transmitted OSPFv3 LS Ack packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 LS Ack packets.

## monitor protocol ospfv3 disable packet ls-request

Disables the generation of debug messages related to OSPFv3 link-state request (LSR) packets.

### Syntax

```
monitor protocol ospfv3 disable packet ls-request [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are disabled for OSPFv3 LSR packets at a medium level of detail.

### Parameters

**detail**

Disables detailed debugging of OSPFv3 LSR packets, both sent and received.

**recv**

Disables debugging of received OSPFv3 LSR packets.

**detail**

Disables detailed debugging of received OSPFv3 LSR packets.

**send**

Disables debugging of transmitted OSPFv3 LSR packets.

**detail**

Disables detailed debugging of transmitted OSPFv3 LSR packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 LSR packets.

## monitor protocol ospfv3 disable packet ls-update

Disables the generation of debug messages related to OSPFv3 link-state update (LSU) packets.

### Syntax

```
monitor protocol ospfv3 disable packet ls-update [ detail | rcv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are disabled for OSPFv3 LSU packets at a medium level of detail.

### Parameters

**detail**

Disables detailed debugging of OSPFv3 LSU packets, both sent and received.

**rcv**

Disables debugging of received OSPFv3 LSU packets.

**detail**

Disables detailed debugging of received OSPFv3 LSU packets.

**send**

Disables debugging of transmitted OSPFv3 LSU packets.

**detail**

Disables detailed debugging of transmitted OSPFv3 LSU packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 LSU packets.

## monitor protocol ospfv3 disable packet recv

Disables the generation of debug messages related to OSPFv3 received packets.

### Syntax

```
monitor protocol ospfv3 disable packet recv
```

### Command Default

Debug messages are disabled for OSPFv3 received packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 received packets.

## monitor protocol ospfv3 disable packet send

Disables the generation of debug messages related to OSPFv3 sent packets.

### Syntax

```
monitor protocol ospfv3 disable packet send
```

### Command Default

Debug messages are disabled for OSPFv3 sent packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 sent packets.

## monitor protocol ospfv3 disable route

Disables the generation of debug messages related to OSPFv3 route calculations.

### Syntax

```
monitor protocol ospfv3 disable route [ ase | ia | install | spf ]
```

### Command Default

Debug messages are disabled for OSPFv3 route calculations.

### Parameters

**ase**

Disables detailed debugging of OSPFv3 external route calculations.

**ia**

Disables detailed debugging of OSPFv3 inter-area route calculations.

**install**

Disables detailed debugging of OSPFv3 route installation.

**spf**

Disables detailed debugging of OSPFv3 Shortest Path First (SPF) calculation.

### Modes

Operational mode

### Usage Guidelines

Use this command to disable the generation of debug messages related to OSPFv3 route calculations.

## monitor protocol ospfv3 enable events

Enables the generation of debug messages related to OSPFv3 events.

### Syntax

```
monitor protocol ospfv3 enable events [ abr | asbr | nssa | os | router | vlink ]
```

### Parameters

**abr**

Enables debugging of Area Border Router (ABR) events.

**asbr**

Enables debugging of Autonomous System Boundary Router (ASBR) events.

**nssa**

Enables debugging of not-so-stubby area (NSSA) events.

**os**

Enables debugging of OS-interaction events.

**router**

Enables debugging of other router events.

**vlink**

Enables debugging of virtual-link events.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 events.

## monitor protocol ospfv3 enable ifsm

Enables the generation of debug messages related to OSPFv3 Interface Finite State Machine (IFSM) events.

### Syntax

```
monitor protocol ospfv3 enable ifsm [ events | status | timers ]
```

### Parameters

**events**

Enables debugging of IFSM events.

**status**

Enables debugging of IFSM status information.

**timers**

Enables debugging of IFSM timer information.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 IFSM events.



## monitor protocol ospfv3 enable lsa

Enables the generation of debug messages related to OSPFv3 Link State Advertisements (LSAs).

### Syntax

```
monitor protocol ospfv3 enable lsa [ flooding | generate | install | maxage | refresh ]
```

### Parameters

**flooding**

Enables debugging of OSPFv3 LSA flood events.

**generate**

Enables debugging of OSPFv3 LSA generation.

**install**

Enables debugging of OSPFv3 LSA installation.

**maxage**

Enables debugging of maximum age processing related OSPFv3 LSA.

**refresh**

Enables debugging of OSPFv3 LSA refreshes.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 LSAs.

## monitor protocol ospfv3 enable nfsm

Enables the generation of debug messages related to OSPFv3 Neighbor Finite State Machines (NFSMs).

### Syntax

```
monitor protocol ospfv3 enable nfsm [ events | status | timers ]
```

### Parameters

**events**

Enables debugging of OSPFv3 NFSM events.

**status**

Enables debugging of OSPFv3 NFSM status.

**timers**

Enables debugging of OSPFv3 NFSM timers.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 NFSM.

## monitor protocol ospfv3 enable nsm

Enables the generation of debug messages related to OSPFv3 Network Services Module (NSM).

### Syntax

```
monitor protocol ospfv3 enable nsm [ interface | redistribute ]
```

### Parameters

**interface**

Enables debugging of OSPFv3 NSM interface events.

**redistribute**

Enables debugging of OSPFv3 NSM redistribute events.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 NSM.

## monitor protocol ospfv3 enable packet all

Enables the generation of debug messages related to all OSPFv3 packets that are arriving at and leaving a router.

### Syntax

```
monitor protocol ospfv3 enable packet all [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are enabled for all OSPFv3 packets at a medium level of detail.

### Parameters

**detail**

Enables detailed debugging of all OSPFv3 packets, both sent and received.

**recv**

Enables debugging of all received OSPFv3 packets.

**detail**

Enables detailed debugging of all received OSPFv3 packets.

**send**

Enables debugging of all transmitted OSPFv3 packets.

**detail**

Enables debugging of all transmitted OSPFv3 packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to all OSPFv3 packets that are arriving at and leaving a router.

## monitor protocol ospfv3 enable packet dd

Enables the generation of debug messages related to OSPFv3 Database Description (DD) packets.

### Syntax

```
monitor protocol ospfv3 enable packet dd [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are generated for OSPF DD packets at a medium level of detail.

### Parameters

**detail**

Enables detailed debugging of OSPFv3 DD packets, both sent and received.

**recv**

Enables debugging of received OSPFv3 DD packets.

**detail**

Enables detailed debugging of received OSPFv3 DD packets.

**send**

Enables debugging of transmitted OSPFv3 DD packets.

**detail**

Enables detailed debugging of transmitted OSPFv3 DD packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 DD packets.

## monitor protocol ospfv3 enable packet detail

Enables the generation of debug messages related to OSPFv3 detailed packets.

### Syntax

```
monitor protocol ospfv3 enable packet detail
```

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 detailed packets.

## monitor protocol ospfv3 enable packet hello

Enables the generation of debug messages related to OSPFv3 Hello packets.

### Syntax

```
monitor protocol ospfv3 enable packet hello [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are generated for OSPFv3 Hello packets at a medium level of detail.

### Parameters

**detail**

Enables detailed debugging of OSPFv3 Hello packets, both sent and received.

**recv**

Enables detailed debugging of received OSPFv3 Hello packets.

**detail**

Enables detailed debugging of received OSPFv3 Hello packets.

**send**

Enables debugging of transmitted OSPFv3 Hello packets.

**detail**

Enables detailed debugging of transmitted OSPFv3 Hello packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 Hello packets.

## monitor protocol ospfv3 enable packet ls-ack

Enables the generation of debug messages related to OSPFv3 link-state acknowledgment (LS Ack) packets.

### Syntax

```
monitor protocol ospfv3 enable packet ls-ack [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are generated for OSPFv3 LS Ack packets at a medium level of detail.

### Parameters

**detail**

Enables detailed debugging of OSPFv3 LS Ack packets, both sent and received.

**recv**

Enables debugging of received OSPFv3 LS Ack packets.

**detail**

Enables detailed debugging of received OSPFv3 LS Ack packets.

**send**

Enables debugging of transmitted OSPFv3 LS Ack packets.

**detail**

Enables detailed debugging of transmitted OSPFv3 LS Ack packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 LS Ack packets.



## monitor protocol ospfv3 enable packet ls-request

Enables the generation of debug messages related to OSPFv3 link-state request (LSR) packets.

### Syntax

```
monitor protocol ospfv3 enable packet ls-request [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are generated for OSPFv3 LSR packets at a medium level of detail.

### Parameters

**detail**

Enables detailed debugging of OSPFv3 LSR packets, both sent and received.

**recv**

Enables debugging of received OSPFv3 LSR packets.

**detail**

Enables detailed debugging of received OSPFv3 LSR packets.

**send**

Enables debugging of transmitted OSPFv3 LSR packets.

**detail**

Enables detailed debugging of transmitted OSPFv3 LSR packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 LSR packets.

## monitor protocol ospfv3 enable packet ls-update

Enables the generation of debug messages related to OSPFv3 link-state update (LSU) packets.

### Syntax

```
monitor protocol ospfv3 enable packet ls-update [ detail | recv [ detail ] | send [ detail ] ]
```

### Command Default

Debug messages are generated for OSPFv3 LSU packets at a medium level of detail.

### Parameters

**detail**

Enables detailed debugging of OSPFv3 LSU packets, both sent and received.

**recv**

Enables debugging of received OSPFv3 LSU packets.

**detail**

Enables detailed debugging of received OSPFv3 LSU packets.

**send**

Enables debugging of transmitted OSPFv3 LSU packets.

**detail**

Enables detailed debugging of transmitted OSPFv3 LSU packets.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 LSU packets.

## monitor protocol ospfv3 enable packet recv

Enables the generation of debug messages related to OSPFv3 received packets.

### Syntax

```
monitor protocol ospfv3 enable packet recv
```

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 received packets.

## monitor protocol ospfv3 enable packet send

Enables the generation of debug messages related to OSPFv3 sent packets.

### Syntax

```
monitor protocol ospfv3 enable packet send
```

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 sent packets.

## monitor protocol ospfv3 enable route

Enables the generation of debug messages related to OSPFv3 route calculations.

### Syntax

```
monitor protocol ospfv3 enable route [ ase | ia | install | spf ]
```

### Parameters

**ase**

Enables detailed debugging of OSPFv3 external route calculations.

**ia**

Enables detailed debugging of OSPFv3 inter-area route calculations.

**install**

Enables detailed debugging of OSPFv3 route installation.

**spf**

Enables detailed debugging of OSPFv3 Shortest Path First (SPF) calculation.

### Modes

Operational mode

### Usage Guidelines

Use this command to enable the generation of debug messages related to OSPFv3 route calculations.

## protocols ospfv3 abr-type

Sets an OSPFv3 Area Border Router (ABR) type.

### Syntax

**set protocols ospfv3** [ *process process-id* ] **abr-type** { *cisco* | *ibm* | *standard* }

**delete protocols ospfv3** [ *process process-id* ] **abr-type**

### Command Default

The default ABR type is cisco.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      abr-type {
        cisco
        ibm
        standard
      }
    }
  }
}
```

### Usage Guidelines

Sets an OSPFv3 ABR type.

Specifying the ABR type allows better functioning in a multi-vendor environment. The following ABR types are supported:

- Cisco (RFC 3509): A router is considered an ABR if it has more than one area actively attached and one of them is the backbone area.
- IBM (RFC 3509): A router is considered an ABR if it has more than one area actively attached and the backbone area is configured. In this case, the configured backbone need not be actively connected.
- Standard (RFC 2328): A router is considered an ABR if it has more than one area actively attached to it.

Use the **set** form of this command to set an OSPFv3 ABR type.

Use the **delete** form of this command to set cisco as the default ABR type.

## protocols ospfv3 auto-cost reference-bandwidth <bandwidth>

Sets the reference bandwidth method for calculating administrative cost.

### Syntax

```
set protocols ospfv3 [ process process-id] auto-cost reference-bandwidth bandwidth
```

```
delete protocols ospfv3 [ process process-id] auto-cost reference-bandwidth
```

```
show protocols ospfv3 [ process process-id] auto-cost reference-bandwidth
```

### Command Default

The default reference bandwidth is 100 Mbps.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*bandwidth*

The reference bandwidth rate in Mbps. The bandwidth ranges from 1 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      auto-cost
      {
        reference-bandwidth bandwidth
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set a reference bandwidth for calculating OSPFv3 administrative cost. The OSPFv3 metric is calculated as the reference bandwidth divided by the actual bandwidth.

An explicitly set cost for an area overrides automatically calculated values.

Use the **set** form of this command to set the reference bandwidth method for calculating OSPFv3 administrative cost.

Use the **delete** form of this command to restore the default reference bandwidth.

Use the **show** form of this command to display the OSPFv3 auto-cost configuration.

## protocols ospfv3 default-information originate

Generates a default external route into an OSPFv3 routing area.

### Syntax

**set protocols ospfv3** [ **process** *process-id* ] **default-information originate** [ **metric** *metric* | **metric-type** *type* ]

**delete protocols ospfv3** [ **process** *process-id* ] **default-information originate** [ **metric** | **metric-type** ]

**show protocols ospfv3** [ **process** *process-id* ] **default-information originate** [ **metric** | **metric-type** ]

### Command Default

By default, the system does not generate a default external route into an OSPFv3 routing area. When enabled to do so, the default depends on the type of area into which the default route is being advertised:

- In stub areas, a Type 3 LSA is generated with a metric of 1 and the metric type is ignored.
- In not-so-stubby areas (NSSAs) configured to import summary advertisements, a Type 7 LSA with a metric of 1 and a metric type of 2 is generated.
- In NSSAs configured not to import summary advertisements, a Type 3 LSA with a metric of 1 and the metric type is ignored.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The external route metric type to be associated with the Type 5 default link-state advertisement (LSA). Supported values are as follows:

**1:** Type 1 external route.

**2:** Type 2 external route.

The default is 2.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      default-information {
        originate
        {
          metric metric
          metric-type type
        }
      }
    }
  }
}
```



```
}  
}
```

## Usage Guidelines

Use this command to generate a default external route into an OSPFv3 routing area. The default route is redistributed into an OSPFv3 routing area.

If you redistribute routes in this way, the router automatically becomes an Autonomous System Boundary Router (ASBR). The router must have a default route configured before it can generate one, unless the **always** keyword is specified.

Use the **set** form of this command to generate a default external route into an OSPFv3 routing area.

Use the **delete** form of this command to disable the generation of a default external route into the OSPFv3 routing area or to restore default parameter values.

Use the **show** form of this command to display default route distribution configuration.

### NOTE

When extended IPv4 ACLs are used in a route map, the route must be specified in the source field of the ACL and the net mask of the route must be specified in the destination field of the ACL.

The inverse mask for the source field is applied to the prefix before the comparison with the source. The inverse mask for the destination field is applied to the net mask of the prefix before the comparison with the net mask as specified in the destination field.

Effectively, the prefix must match the source to the application of the inverse mask of the source field. And the net mask for the prefix must match the destination to the application of the inverse mask of the destination field.

## protocols ospfv3 default-metric <metric>

Sets the default metric to be applied to routes from other protocols being redistributed into OSPFv3.

### Syntax

```
set protocols ospfv3 [ process process-id] default-metric metric
```

```
delete protocols ospfv3 [ process process-id] default-metric
```

```
show protocols ospfv3 [ process process-id] default-metric
```

### Command Default

Built-in, automatic metric translations, as appropriate for each routing protocol.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The metric to be applied to routes from other protocols being redistributed into OSPFv3. The metric ranges from 0 to 16777214.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      default-metric metric
    }
  }
}
```

### Usage Guidelines

Use this command to set the default metric to be applied to routes from other protocols being redistributed into OSPFv3.

Use the **set** form of this command to set the default OSPFv3 metric.

Use the **delete** form of this command to restore the default metric.

Use the **show** form of this command to display the OSPFv3 default metric configuration.

## protocols ospfv3 distance

Sets the administrative distance for OSPFv3 routes.

### Syntax

```
set protocols ospfv3 [ process process-id] distance [ global global| ospfv3 [ external external| inter-area inter| intra-area intra ]]
```

```
delete protocols ospfv3 [ process process-id] distance [ global global| ospfv3 [ external | inter-area | intra-area ]]
```

```
show protocols ospfv3 [ process process-id] distance [ global global| ospfv3 [ global global| ospfv3 [ external | inter-area | intra-area ]]
```

### Command Default

The default administrative distance for OSPFv3 routes is 110.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

**global**

The administrative distance for all routes. The distance ranges from 1 to 254.

*external*

The OSPFv3 administrative distance for external routes (routes learned from another protocol by redistribution). The distance ranges from 1 to 254. The default is 110.

*inter*

The OSPFv3 administrative distance for inter-area routes (routes to another area). The distance ranges from 1 to 254. The default distance is 110.

*intra*

The OSPFv3 administrative distance for intra-area routes (routes within an area). The distance ranges from 1 to 254. The default distance is 110.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      distance {
        global global
        ospfv3 {
          external external
          inter-area inter
          intra-area intra
        }
      }
    }
  }
}
```

```
}  
}
```

## Usage Guidelines

Use this command to set the administrative distance for OSPFv3 routes.

The administrative distance indicates the trustworthiness of a router or group of routers as a source of routing information. In general, the higher the distance, the less trusted the entity. An administrative distance of 1 usually represents a directly connected network, and an administrative distance of 255 means the routing source is unreliable or unknown. The administrative distance conventionally applied to OSPFv3 is 110.

Use the **set** form of this command to set the administrative distance for OSPFv3 routes.

Use the **delete** form of this command to restore the default administrative distance.

Use the **show** form of this command to display the administrative distance configuration.

## protocols ospfv3 distribute-list <name>

Specifies the OSPFv3 access list used to filter networks in routing updates.

### Syntax

```
set protocols ospfv3 distribute-list name [ export type ] [ import ]
delete protocols ospfv3 distribute-list name [ export type | import ]
show protocols ospfv3 distribute-list name
```

### Parameters

*name*

The name of the access list that is used to filter networks in routing updates.

**export** *type*

The type of export routes to filter. Possible values include: **bgp**, **connected**, **kernel**, **rip**, or **static**. Multiple types can be specified by creating additional **export** configuration nodes.

**import**

The filter for incoming routing updates.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    distribute-list name{
      export type
      import
    }
  }
}
```

### Usage Guidelines

Use this command to specify an access list to filter networks in routing updates.

Use the **set** form of this command to specify an access list.

Use the **delete** form of this command to remove an access list.

Use the **show** form of this command to display the configuration.

## protocols ospfv3 log

Enables OSPFv3 protocol logs.

### Syntax

```
set protocols ospfv3 log { all | events | ifsm | lsa | nfsm | nsm | packet | rib | route }
delete protocols ospfv3 log { all | events | ifsm | lsa | nfsm | nsm | packet | rib | route }
show protocols ospfv3 log { all | events | ifsm | lsa | nfsm | nsm | packet | rib | route }
```

### Command Default

None

### Parameters

**all**  
Enables all OSPFv3 logs.

**events**  
Enables only OSPFv3 event logs.

**ifsm**  
Enables only OSPFv3 IFSM logs.

**lsa**  
Enables only OSPFv3 LSA logs.

**nfsm**  
Enables only OSPFv3 NFSM logs.

**nsm**  
Enables only OSPFv3 NSM logs.

**packet**  
Enables only OSPFv3 packet logs.

**rib**  
Enables only OSPFv3 RIB logs.

**route**  
Enables only OSPFv3 route logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      all
      events
      ifsm
    }
  }
}
```

```
    lsa
    nfsm
    nsm
    packet
    rib
    route
  }
}
```

## Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First (OSPF)v3 logs.

Use the **delete** form of this command to remove OSPFv3 logs.

Use the **show** form of this command to view OSPFv3 logs.

## protocols ospfv3 log events

Enables OSPFv3 event logs.

### Syntax

```
set protocols ospfv3 log events { abr | all | asbr | lsa | nssa | os | router | vlink }
delete protocols ospfv3 log events { abr | all | asbr | lsa | nssa | os | router | vlink }
show protocols ospfv3 log events { abr | all | asbr | lsa | nssa | os | router | vlink }
```

### Command Default

None

### Parameters

<b>abr</b>	Enables OSPFv3 event ABR logs.
<b>all</b>	Enables all OSPFv3 event logs
<b>asbr</b>	Enables only OSPFv3 event ASBR logs.
<b>lsa</b>	Enables only OSPFv3 event LSA logs.
<b>nssa</b>	Enables only OSPFv3 event NSSA logs.
<b>os</b>	Enables only OSPFv3 event OS logs.
<b>router</b>	Enables only OSPFv3 event router logs.
<b>vlink</b>	Enables only OSPFv3 event vlink logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      events {
        abr
        all
        asbr
        lsa
        nssa
      }
    }
  }
}
```



```
        ospf
        router
        vlink
    }
}
}
```

## Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First (OSPF) event logs.

Use the **delete** form of this command to remove OSPFv3 event logs.

Use the **show** form of this command to view OSPFv3 event logs.

## protocols ospfv3 log ifsm

Enables OSPFv3 IFSM logs.

### Syntax

```
set protocols ospfv3 log ifsm { all | events | status | timers}
delete protocols ospfv3 log ifsm { all | events | status | timers}
show protocols ospfv3 log ifsm { all | events | status | timers}
```

### Command Default

None

### Parameters

**all**  
Enables all OSPFv3 IFSM logs.

**events**  
Enables only OSPFv3 IFSM event logs.

**status**  
Enables only OSPFv3 IFSM status logs.

**timers**  
Enables only OSPFv3 IFSM timer logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      ifsm {
        all
      }
      events
      status
      timers
    }
  }
}
```

### Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First (OSPF) interface finite-state machine (IFSM) logs.

Use the **delete** form of this command to remove OSPFv3 IFSM logs.

Use the **show** form of this command to view OSPFv3 IFSM logs.

## protocols ospfv3 log lsa

Enables OSPFv3 LSA logs.

### Syntax

```
set protocols ospfv3 log lsa { all | flooding | generate | install | maxage | refresh}
delete protocols ospfv3 log lsa { all | flooding | generate | install | maxage | refresh}
show protocols ospfv3 log lsa { all | flooding | generate | install | maxage | refresh}
```

### Command Default

None

### Parameters

**all**  
Enables all OSPFv3 LSA logs.

**flooding**  
Enables only OSPFv3 LSA flooding logs.

**generate**  
Enables only OSPFv3 LSA generation logs.

**install**  
Enables only OSPFv3 LSA installation logs.

**maxage**  
Enables only OSPFv3 LSA maximum age logs.

**refresh**  
Enables only OSPFv3 LSA refresh logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      lsa {
        all
        flooding
        generate
        install
        maxage
        refresh
      }
    }
  }
}
```

## Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First (OSPF)v3 LSA logs.

Use the **delete** form of this command to remove OSPFv3 LSA logs.

Use the **show** form of this command to view OSPFv3 LSA logs.

## protocols ospfv3 log nfsm

Enables OSPFv3 NFSM logs.

### Syntax

```
set protocols ospfv3 log nfsm { all | events | status | timers }
```

```
delete protocols ospfv3 log nfsm { all | events | status | timers }
```

```
show protocols ospfv3 log nfsm { all | events | status | timers }
```

### Command Default

None

### Parameters

#### all

Enables all OSPFv3 NFSM logs.

#### events

Enables only OSPFv3 NFSM event logs.

#### status

Enables only OSPFv3 NFSM status logs.

#### timers

Enables only OSPFv3 NFSM timers logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      nfsm {
        all
        events
        status
        timers
      }
    }
  }
}
```

### Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First (OSPF)v3 neighbor FSM (NFSM) logs.

Use the **delete** form of this command to remove OSPFv3 NFSM logs.

Use the **show** form of this command to view OSPFv3 NFSM logs.

## protocols ospfv3 log nsm

Enables OSPFv3 NSM logs.

### Syntax

```
set protocols ospfv3 nsm { all | interface | redistribute}
delete protocols ospfv3 nsm { all | interface | redistribute}
show protocols ospfv3 nsm { all | interface | redistribute}
```

### Command Default

None

### Parameters

**all**  
Enables all OSPF NSM logs.

**interface**  
Enables only OSPF interface logs.

**redistribute**  
Enables only OSPF redistribute logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      nsm {
        all
        interface
        redistribute
      }
    }
  }
}
```

### Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First v3 (OSPFv3) network services module (NSM) log.

Use the **delete** form of this command to remove an OSPFv3 NSM log.

Use the **show** form of this command to remove an OSPFv3 NSM log.

## protocols ospfv3 log packet

Enables OSPFv3 packet logs.

### Syntax

```
set protocols ospfv3 log packet { all | dd | detail| hello| ls-ack| ls-request| ls-update| recv| send}
delete protocols ospfv3 log packet { all | dd | detail| hello| ls-ack| ls-request| ls-update| recv| send}
show protocols ospfv3 log packet { all | dd | detail| hello| ls-ack| ls-request| ls-update| recv| send}
```

### Command Default

None

### Parameters

**all**  
Enables all OSPFv3 packet logs.

**dd**  
Enables only OSPFv3 packet DD logs.

**detail**  
Enables only OSPFv3 packet detail logs.

**hello**  
Enables only OSPFv3 packet hello logs.

**ls-ack**  
Enables only OSPFv3 packet ls-ack logs.

**ls-request**  
Enables only OSPFv3 packet ls-request logs.

**ls-update**  
Enables only OSPFv3 packet ls-update logs.

**recv**  
Enables only OSPFv3 packet receive logs.

**send**  
Enables only OSPFv3 packet send logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      packet {
        all
        dd
```

```
    detail
    hello
    ls-ack
ls-request
    ls-update
        recv
        send
    }
}
}
```

## Usage Guidelines

Use the **set** form of this command to enable Open Shortest Path First (OSPF)v3 packet logs.

Use the **delete** form of this command to remove OSPFv3 packet logs.

Use the **show** form of this command to view OSPFv3 packet logs.



## protocols ospfv3 log rib

Enables OSPFv3 RIB logs.

### Syntax

```
set protocols ospfv3 log rib { all | interface | redistribute}
delete protocols ospfv3 log rib { all | interface | redistribute}
show protocols ospfv3 log rib { all | interface | redistribute}
```

### Command Default

None

### Parameters

**all**  
Enables all OSPFv3 RIB logs.

**interface**  
Enables only OSPFv3 RIB interface logs.

**redistribute**  
Enables only OSPFv3 RIB redistribute logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    log {
      rib {
        all
        interface
        redistribute
      }
    }
  }
}
```

### Usage Guidelines

Use the **set** form of this command to enable OSPFv3 RIB logs.

Use the **delete** form of this command to remove OSPFv3 RIB logs.

Use the **show** form of this command to view OSPFv3 RIB logs.

## protocols ospfv3 log route

Enables OSPFv3 route logs.

### Syntax

```
set protocols ospfv3 log route { all | ase | ia | install | spf }
delete protocols ospfv3 log route { all | ase | ia | install | spf }
show protocols ospfv3 log route { all | ase | ia | install | spf }
```

### Command Default

None

### Parameters

**all**  
Enables all OSPFv3 route all logs.

**ase**  
Enables only OSPFv3 route ASE logs.

**ia**  
Enables only OSPFv3 route IA logs.

**install**  
Enables only OSPFv3 route install logs.

**SPF**  
Enables only OSPFv3 route SPF logs.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospf {
    log {
      route {
        all
        ase
        ia
        install
        spf
      }
    }
  }
}
```

### Usage Guidelines

Use the **set** form of this command to enable OSPFv3 route logs.

Use the **delete** form of this command to remove OSPFv3 route logs.

Use the **show** form of this command to view OSPFv3 route logs.

## protocols ospfv3 passive-interface

Suppresses routing updates on an interface.

### Syntax

```
set protocols ospfv3 [ process process-id] passive-interface { default | interface }
delete protocols ospfv3 [ process process-id] passive-interface { default | interface }
show protocols ospfv3 [ process process-id] passive-interface
```

### Command Default

Routing updates are not suppressed.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

**default**

Routing updates are to be suppressed on all interfaces by default.

*interface*

The identifier of an interface. Supported interface types are:

- Data plane
- Loopback

For more information about these interface types, refer to [Supported Interface Types](#) on page 181.

You may suppress routing updates on a per-interface basis.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      passive-interface [default | interface]
    }
  }
}
```

### Usage Guidelines

Use this command to specify suppression for OSPFv3 routing updates on all interfaces (using **default**) or a specific interface. OSPFv3 traffic can be received on the interface but it will not be sent on it.

Use the **set** form of this command to suppress routing updates for an interface.

Use the **delete** form of this command to remove routing update suppression.

Use the **show** form of this command to display routing update suppression configuration.

## protocols ospfv3 passive-interface-exclude

Excludes the suppression of the OSPFv3 routing updates on an interface.

### Syntax

**set** protocols ospf passive-interface-exclude *interface*

**delete** protocols ospf passive-interface-exclude *interface*

**show** protocols ospf passive-interface-exclude

### Command Default

Routing updates are not suppressed and **passive-interface** is configured.

### Configuration Statement

```
protocols {
  ospfv3 {
    passive-interface-exclude interface
  }
}
```

### Parameters

*interface*

The identifier of an interface. Supported interface types are:

- Data plane
- Loopback

For more information about these interface types, refer to [Supported Interface Types](#) on page 181.

You can exclude the suppression of routing updates on multiple interfaces by creating multiple **passive-interface-exclude** configuration nodes.

### Modes

Configuration mode

### Usage Guidelines

Use this command to exclude the suppression of OSPFv3 routing updates on an interface. OSPFv3 traffic can be neither received on the interface nor sent through the interface.

Use the **set** form of this command to exclude the suppression of OSPFv3 routing updates on an interface.

Use the **delete** form of this command to restore the default behavior, that is, routing updates are not suppressed.

Use the **show** form of this command to display the suppression configuration of OSPFv3 routing updates.

## Examples

The following example shows how to exclude the suppression of OSPF routing update on the 10.11.121.134 interface.

```
vyatta@Rn# set protocols ospf passive-interface-exclude 10.11.121.134
vyatta@R6# show protocols ospfv3
ospf {
    passive-interface-exclude 10.11.121.134
}
```

## protocols ospfv3 process <process-id> address-family ipv4 unicast

Configures IPv4 unicast addresses for OSPFv3 per address-family mode.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast
delete protocols ospfv3 process process-id address-family ipv4 unicast
show protocols ospfv3 process process-id address-family ipv4 unicast
```

### Parameters

*process-id*  
An OSPFv3 process ID.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process-id process-id {
      address-family {
        ipv4 {
          unicast {
            }
          }
        }
      }
    }
  }
}
```

### Usage Guidelines

This command allows you to enter address family mode in which you can configure IPv4 unicast addresses for OSPFv3, which includes the following tasks:

- Summarize intra-area IPv4 routes (see [protocols ospfv3 process <process-id> address-family ipv4 unicast area <area-id> range](#) on page 81).
- Create a default external route (see [protocols ospfv3 process <process-id> address-family ipv4 unicast default-information originate](#) on page 83).
- Redistribute IPv4 routes (see [protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute](#) on page 85).
- Summarize IPv4 external routes (see [protocols ospfv3 process <process-id> address-family ipv4 unicast summary-address](#) on page 97).

RFC 5838 defines the range of the following instance IDs to use for each address family in OSPFv3.

TABLE 2

Instance ID#	Address Family
0 - 31	IPv6 unicast
64 - 95	IPv4 unicast

Multiple router processes can be configured on each interface. Each instance ID creates a separate OSPFv3 instance with its own neighbor adjacencies, link state database, and SPF computation. You must configure at least two instances: one base and one named instance, of which each must be configured with a different address family.

**NOTE**

The Brocade vRouter does not support running a single IPv4 or IPv6 OSPFv3 process on multiple instances on the same interface.

Use the **set** form of this command to enter address family mode in which you can configure IPv4 unicast addresses for OSPFv3.

Use the **delete** form of this command to remove the IPv4 address family configuration for OSPFv3.

Use the **show** form of this command to display the IPv4 unicast addresses for OSPFv3.



## protocols ospfv3 process <process-id> address-family ipv4 unicast area <area-id> range

Configures a range of OSPFv3 addresses.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast area area-id range { ipv4net | ipv6net } [ advertise | not-advertise ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast area area-id range { ipv4net | ipv6net } [ advertise | not-advertise ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast area area-id range { ipv4net | ipv6net } [ advertise | not-advertise ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967295.

*ipv4net*

A range to be summarized, expressed as an IPv4 network in the format *ipv4-address/prefix*.

*ipv6net*

A range to be summarized, expressed as an IPv6 network in the format *ipv6-address/prefix*.

**advertise**

Advertises this range.

**not-advertise**

Does not advertise this range. Directs the router not to advertise routes in this range.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 {
          unicast {
            area area-id {
              range {
                ipv4net {
                  advertise
                  not-advertise
                }
                ipv6net {
                  advertise
                  not-advertise
                }
              }
            }
          }
        }
      }
    }
  }
}
```



## protocols ospfv3 process <process-id> address-family ipv4 unicast default-information originate

Generates a default external route into an OSPFv3 routing area.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast default-information originate [ always | metric metric |
metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast default-information originate [ always | metric | metric-
type | route-map ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast default-information originate [ always | metric | metric-
type | route-map ]
```

### Command Default

By default, the system does not generate a default external route into an OSPFv3 routing area.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

**always**

Advertises the default route regardless of whether there is a default route.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          default-information {
            originate {
              always
              metric metric
            }
          }
        }
      }
    }
  }
}
```



## protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute

Configures redistribute information from another routing protocol.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast redistribute [ bgp | connected | kernel | rip | static ] [ metric metric | metric-type type route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast redistribute [ bgp | connected | kernel | rip | static ] [ metric metric | metric-type type route-map map-name ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast redistribute [ bgp | connected | kernel | rip | static ] [ metric metric | metric-type type route-map map-name ]
```

### Command Default

By default, only the IPv6 unicast address family is supported. Support for IPv4 unicast may be configured using this command.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

**bgp**

Specifies BGP routes.

**connected**

Specifies connected routes.

**kernel**

Specifies kernel routes.

**static**

Specifies static routes.

**rip**

Specifies Routing Information Protocol (RIP) routes.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

## Modes

Configuration mode

## Configuration Statement

```

protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          redistribute {
            bgp {
              metric metric
              metric-type type
              route-map map-name
            }
            connected {
              metric metric
              metric-type type
              route-map map-name
            }
            kernel {
              metric metric
              metric-type type
              route-map map-name
            }
            rip {
              metric metric
              metric-type type
              route-map map-name
            }
            static {
              metric metric
              metric-type type
              route-map map-name
            }
          }
        }
      }
    }
  }
}

```

## Usage Guidelines

Use this command to import IPv4 routes from other routing protocols, or from another OSPFv3 instance, into OSPFv3 AS-external-LSAs.

### NOTE

This command applies to IPv4 unicast routes only.

OSPFv3 advertises routes learned from other routing protocols or from other OSPFv3 instances, including static or connected routes. Each injected prefix is put into the AS-external-LSA with a specified metric and metric-type.

Use the **set** form of this command to import IPv4 routes from other routing protocols or from another OSPFv3 instance, into OSPFv3 AS-external-LSAs.

Use the **delete** form of this command to restore to the default parameter values.

Use the **show** form of this command to display redistribute information.

## protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute bgp

Sets the parameters for redistribution of Border Gateway Protocol (BGP) routes into OSPFv3.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast redistribute bgp [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast redistribute bgp [ metric metric | metric-type type | route-map map-name ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast redistribute bgp [ metric metric | metric-type type | route-map map-name ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          redistribute {
            bgp {
              metric metric
              metric-type type
              route-map map-name
            }
          }
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to define the parameters for redistribution of BGP routes into OSPFv3.

### NOTE

This command applies to IPv4 unicast routes only.

Use the **set** form of this command to set BGP route redistribution parameters.

Use the **delete** form of this command to remove BGP route redistribution parameters.

Use the **show** form of this command to display BGP route redistribution configuration.

### NOTE

When extended IPv4 ACLs are used in a route map, the route must be specified in the source field of the ACL and the net mask of the route must be specified in the destination field of the ACL.

The inverse mask for the source field is applied to the prefix before the comparison with the source. The inverse mask for the destination field is applied to the net mask of the prefix before the comparison with the net mask as specified in the destination field.

Effectively, the prefix must match the source to the application of the inverse mask of the source field. And the net mask for the prefix must match the destination to the application of the inverse mask of the destination field.



## protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute connected

Sets the parameters for redistribution of connected routes into OSPFv3.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast redistribute connected [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast redistribute connected [ metric | metric-type | route-map ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast redistribute connected [ metric metric | metric-type type | route-map map-name ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          redistribute {
            connected {
              metric metric
              metric-type type
              route-map map-name
            }
          }
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to specify the parameters for redistribution of connected routes into OSPFv3.

### NOTE

This command applies to IPv4 unicast routes only.

Use the **set** form of this command to set connected route redistribution parameters.

Use the **delete** form of this command to remove connected route redistribution parameters.

Use the **show** form of this command to display connected route redistribution configuration.

### NOTE

When extended IPv4 ACLs are used in a route map, the route must be specified in the source field of the ACL and the net mask of the route must be specified in the destination field of the ACL.

The inverse mask for the source field is applied to the prefix before the comparison with the source. The inverse mask for the destination field is applied to the net mask of the prefix before the comparison with the net mask as specified in the destination field.

Effectively, the prefix must match the source to the application of the inverse mask of the source field. And the net mask for the prefix must match the destination to the application of the inverse mask of the destination field.

## protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute kernel

Sets the parameters for redistribution of kernel routes into OSPFv3.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast redistribute kernel [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast redistribute kernel [ metric metric | metric-type type | route-map map-name ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast redistribute kernel [ metric metric | metric-type type | route-map map-name ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          redistribute {
            kernel {
              metric metric
              metric-type type
              route-map map-name
            }
          }
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to specify the parameters for redistribution of kernel routes into OSPFv3.

### NOTE

This command applies to IPv4 unicast routes only.

Use the **set** form of this command to set kernel route redistribution parameters.

Use the **delete** form of this command to remove kernel route redistribution parameters.

Use the **show** form of this command to display kernel route redistribution configuration.

### NOTE

When extended IPv4 ACLs are used in a route map, the route must be specified in the source field of the ACL and the net mask of the route must be specified in the destination field of the ACL.

The inverse mask for the source field is applied to the prefix before the comparison with the source. The inverse mask for the destination field is applied to the net mask of the prefix before the comparison with the net mask as specified in the destination field.

Effectively, the prefix must match the source to the application of the inverse mask of the source field. And the net mask for the prefix must match the destination to the application of the inverse mask of the destination field.

## protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute rip

Sets the parameters for redistribution of Routing Information Protocol (RIP) routes into OSPFv3.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast redistribute rip [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast redistribute rip [ metric | metric-type | route-map ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast redistribute rip [ metric | metric-type | route-map ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4net*

A range to be summarized, expressed as an IPv4 network in the format *ipv4-address/prefix*.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          redistribute {
            rip {
              metric metric
              metric-type type
              route-map map-name
            }
          }
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to specify the parameters for redistribution of RIP routes into OSPFv3.

### NOTE

This command applies to IPv4 unicast routes only.

Use the **set** form of this command to set the RIP route redistribution parameters.

Use the **delete** form of this command to remove RIP route redistribution parameters.

Use the **show** form of this command to display RIP route redistribution configuration.

### NOTE

When extended IPv4 ACLs are used in a route map, the route must be specified in the source field of the ACL and the net mask of the route must be specified in the destination field of the ACL.

The inverse mask for the source field is applied to the prefix before the comparison with the source. The inverse mask for the destination field is applied to the net mask of the prefix before the comparison with the net mask as specified in the destination field.

Effectively, the prefix must match the source to the application of the inverse mask of the source field. And the net mask for the prefix must match the destination to the application of the inverse mask of the destination field.

## protocols ospfv3 process <process-id> address-family ipv4 unicast redistribute static

Sets the parameters for redistribution of static routes into OSPFv3.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast redistribute static [ metric metric | metric-type type |
route-map map-name ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast redistribute static [ metric metric | metric-type type |
route-map map-name ]
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast redistribute static [ metric metric | metric-type type |
route-map map-name ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The OSPFv3 address-family metric to be applied to the default route. The range is 0 to 16777214. The default is 1.

*type*

The OSPFv3 metric type for address-family default routes. Supported values are as follows:

1: Type 1 external route.

2: Type 2 external route.

The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 unicast {
          redistribute {
            static {
              metric metric
              metric-type type
              route-map map-name
            }
          }
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to specify the parameters for redistribution of static routes into OSPFv3.

### NOTE

This command applies to IPv4 unicast routes only.

Use the **set** form of this command to set the static route redistribution parameters.

Use the **delete** form of this command to remove the static route redistribution parameters.

Use the **show** form of this command to display the configuration of the static route redistribution.

### NOTE

When extended IPv4 ACLs are used in a route map, the route must be specified in the source field of the ACL and the net mask of the route must be specified in the destination field of the ACL.

The inverse mask for the source field is applied to the prefix before the comparison with the source. The inverse mask for the destination field is applied to the net mask of the prefix before the comparison with the net mask as specified in the destination field.

Effectively, the prefix must match the source to the application of the inverse mask of the source field. And the net mask for the prefix must match the destination to the application of the inverse mask of the destination field.



## protocols ospfv3 process <process-id> address-family ipv4 unicast summary-address

Summarizes or suppresses external routes with a specified address range.

### Syntax

```
set protocols ospfv3 process process-id address-family ipv4 unicast summary-address { ipv6net | ipv4net } [ non-advertise | tag tag-value ]
```

```
delete protocols ospfv3 process process-id address-family ipv4 unicast summary-address { ipv6net | ipv4net }
```

```
show protocols ospfv3 process process-id address-family ipv4 unicast summary-address { ipv6net | ipv4net }
```

### Command Default

Routing updates are not suppressed.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv6net*

A range to be summarized, expressed as an IPv6 network in the format *ipv6-address/prefix*.

**not-advertise**

Suppresses routes that match a range.

**tag-value**

Tag value to use as a "match" value for controlling redistribution via route maps. Set a tag value. The tag value ranges from 0 to 4294967295. The default is 0.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      address-family {
        ipv4 {
          unicast {
            summary-address {
              ipv4net {
                non-advertise
                tag tag-value}
              ipv6net {
                non-advertise
                tag tag-value}
            }
          }
        }
      }
    }
  }
}
```

```
}  
}
```

## Usage Guidelines

Use this command to summarize or suppress external routes with a specified address range.

### NOTE

This command applies to IPv4 unicast routes only.

Use the **set** form of this command to summarize or suppress external routes with a specified address range.

Use the **delete** form of this command to remove summary addresses.

Use the **show** form of this command to display summary-addresses configuration.

## protocols ospfv3 process <process-id> passive-interface-exclude

Excludes the suppression of OSPFv3 routing updates on an interface that belongs to the OSPF process.

### Syntax

```
set protocols ospfv3 process [ process process-id] passive-interface-exclude interface
delete protocols ospfv3 process [ process process-id] passive-interface-exclude interface
show protocols ospfv3 process [ process process-id] passive-interface-exclude
```

### Command Default

Routing updates are not suppressed and **passive-interface** is configured.

### Parameters

*interface*

The identifier of an interface. Supported interface types are:

- Data plane
- Loopback

For more information about these interface types, refer to [Supported Interface Types](#) on page 181.

You can exclude the suppression of routing updates on multiple interfaces by creating multiple **passive-interface-exclude** configuration nodes.

### Modes

Configuration Mode

### Configuration Statement

```
protocols {
    ospfv3 {
        process process-id {
            passive-interface-exclude interface
        }
    }
}
```

### Usage Guidelines

Use this command to exclude the suppression of OSPFv3 routing updates on an interface that belongs to the OSPFv3 process. OSPFv3 traffic can be neither received on the interface nor sent through the interface that belongs to the OSPFv3 process.

Use the **set** form of this command to exclude the suppression of OSPFv3 routing updates on an interface that belongs to the OSPFv3 process.

Use the **delete** form of this command to restore the default behavior, that is, routing updates are not suppressed.

Use the **show** form of this command to display the suppression configuration of OSPFv3 routing updates.

## Examples

The following example shows how to exclude the OSPFv3 routing update suppression on 10.11.121.134 interface.

```
vyatta@Rn# set protocols ospf passive-interface-exclude 10.11.121.134 that belong to an OSPF process
with process ID as 12.
vyatta@R6# show protocols ospfv3 process 12
ospf {
    process 12
    {
        passive-interface-exclude 10.11.121.134
    }
}
```

## protocols ospfv3 redistribute bgp

Sets the parameters for redistribution of Border Gateway Protocol (BGP) routes into OSPFv3.

### Syntax

```
set protocols ospfv3 [ process process-id] redistribute bgp [ metric metric | metric-type type | route-map map-name | tag tag-value ]
```

```
delete protocols ospfv3 [ process process-id] redistribute bgp [ metric | metric-type | route-map | tag ]
```

```
show protocols ospfv3 [ process process-id] redistribute bgp [ metric | metric-type | route-map | tag ]
```

### Command Default

Connected routes being redistributed into OSPFv3 are assigned a routing metric of 1 and a metric-type of 2. By default, no route map is applied to redistributed connected routes. The default tag value is 0.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The specified metric to connected routes being redistributed into OSPFv3. The range is 1 to 16. The default is 1.

*type*

The type of calculated cost. Supported values are 1 (internal cost is added to external cost) and 2 (only external cost is used). The default is 2.

*map-name*

A route map for which routes are redistributed.

*tag-value*

Tag value to use as a "match" value for controlling redistribution via route maps. Set a tag value. The tag value ranges from 0 to 2147483647. The default is 0.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      redistribute {
        bgp {
          metric metric
          metric-type type
          route-map map-name
          tag tag-value
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to define the parameters for redistribution of BGP routes into OSPFv3.

Use the **set** form of this command to set BGP route redistribution parameters.

Use the **delete** form of this command to remove BGP route redistribution parameters.

Use the **show** form of this command to display BGP route redistribution configuration.

## protocols ospfv3 redistribute connected

Sets the parameters for redistribution of connected routes into OSPFv3.

### Syntax

```
set protocols ospfv3 [ process process-id] redistribute connected [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 [ process process-id] redistribute connected [ metric | metric-type | route-map ]
```

```
show protocols ospfv3 [ process process-id] redistribute connected [ metric | metric-type | route-map ]
```

### Command Default

Connected routes being redistributed into OSPFv3 are assigned a routing metric of 1 and a metric-type of 2. By default, no route map is applied to redistributed connected routes.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The specified metric to connected routes being redistributed into OSPFv3. The range is 1 to 16. The default is 1.

*type*

The type of calculated cost. Supported values are 1 (internal cost is added to external cost) and 2 (only external cost is used). The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      redistribute {
        connected {
          metric metric
          metric-type type
          route-map map-name
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the parameters for redistribution of connected routes into OSPFv3.

Use the **set** form of this command to set connected route redistribution parameters.

Use the **delete** form of this command to remove connected route redistribution parameters.

Use the **show** form of this command to display connected route redistribution configuration.



## protocols ospfv3 redistribute kernel

Sets the parameter for redistribution of kernel routes into OSPFv3.

### Syntax

**set** protocols ospfv3 [ process *process-id* ] redistribute kernel [ metric *metric* | metric-type *type* | route-map *map-name* ]

**delete** protocols ospfv3 [ process *process-id* ] redistribute kernel [ metric | metric-type | route-map ]

**show** protocols ospfv3 [ process *process-id* ] redistribute kernel [ metric | metric-type | route-map ]

### Command Default

By default, no route map is applied to redistributed kernel routes.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The specified metric to connected routes being redistributed into OSPFv3. The range is 1 to 16. The default is 1.

*type*

The type of calculated cost. Supported values are 1 (internal cost is added to external cost) and 2 (only external cost is used). The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      redistribute {
        kernel {
          metric metric
          metric-type type
          route-map map-name
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify the parameter for redistribution of kernel routes into OSPFv3.

Use the **set** form of this command to set kernel route redistribution parameters.

Use the **delete** form of this command to remove kernel route redistribution parameters.

Use the **show** form of this command to display kernel route redistribution configuration.

## protocols ospfv3 redistribute rip

Sets the parameters for redistribution of RIP next generation (RIPng) routes into OSPFv3.

### Syntax

```
set protocols ospfv3 [ process process-id ] redistribute rip [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 [ process process-id ] redistribute rip [ metric | metric-type | route-map ]
```

```
show protocols ospfv3 [ process process-id ] redistribute rip [ metric | metric-type | route-map ]
```

### Command Default

RIP routes being redistributed into OSPFv3 are assigned a routing metric of 1 and a metric-type of 2. By default, no route map is applied to redistributed RIP routes.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The specified metric to connected routes being redistributed into OSPFv3. The range is 1 to 16. The default is 1.

*type*

The type of calculated cost. Supported values are 1 (internal cost is added to external cost) and 2 (only external cost is used). The default is 2.

*map-name*

A route map for which routes are redistributed.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      redistribute {
        rip {
          metric metric
          metric-type type
          route-map map-name
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify the parameter for redistribution of RIPng routes into OSPFv3.

Use the **set** form of this command to set the RIPng route redistribution parameters.

Use the **delete** form of this command to remove RIPng route redistribution parameters.

Use the **show** form of this command to display RIPng route redistribution configuration.

## protocols ospfv3 redistribute static

Specifies the parameters for redistribution of static routes into OSPFv3.

### Syntax

```
set protocols ospfv3 [ process process-id ] redistribute static [ metric metric | metric-type type | route-map map-name ]
```

```
delete protocols ospfv3 [ process process-id ] redistribute static [ metric | metric-type | route-map ]
```

```
show protocols ospfv3 [ process process-id ] redistribute static [ metric | metric-type | route-map ]
```

### Command Default

Static routes being redistributed into OSPFv3 are assigned a routing metric of 1 and a metric-type of 2. By default, no route map is applied to redistributed static routes.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*metric*

The specified metric to connected routes being redistributed into OSPFv3. The range is 1 to 16. The default is 1.

*type*

The type of calculated cost. Supported values are 1 (internal cost is added to external cost) and 2 (only external cost is used). The default is 2.

*map-name*

Redistributes routes that satisfy the specified route map.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      redistribute {
        static
        {
          metric metric
          metric-type type
          route-map map-name
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify the parameters for redistribution of static routes into OSPFv3.

Use the **set** form of this command to set the static route redistribution parameters.

Use the **delete** form of this command to remove static route redistribution parameters.

Use the **show** form of this command to display static route redistribution configuration.

## protocols ospfv3 router-id <router-id>

Specifies a router identifier (ID) for an OSPFv3 process.

### Syntax

**set protocols ospfv3** [ **process** *process-id*] **router-id** *router-id*

**delete protocols ospfv3** [ **process** *process-id*] **router-id** *router-id*

**show protocols ospfv3** [ **process** *process-id*] **router-id** *router-id*

### Command Default

If no router ID is explicitly configured, the OSPF3 process calculates an ID for the router by using the following algorithm:

1. Use the IP address of the loopback interface.
2. Use the highest IP address of the addresses on router interfaces.
3. If no interfaces are defined, use 0.0.0.0.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*router-id*

The router ID for an OSPFv3 process. This ID overrides the router ID calculated by the OSPFv3 process. The format is an IPv4 address.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      router-id router-id
    }
  }
}
```

### Usage Guidelines

Use this command to specify a router ID for an OSPFv3 process.

Configure each router with a unique router-id. In an OSPFv3 router process that has active neighbors, a new router-id is used at the next reload or when you start the OSPFv3 manually.



#### CAUTION

Modifying the router ID causes the router to restart.

**NOTE**

The minimal requirement for a functioning OSPFv3 setup is to configure both the **protocols ospfv3 router-id** <router-id> command and the **protocols ospfv3 area** <area-id> command. Thereafter, you must commit the configuration.

Use the **set** form of this command to specify a router ID for an OSPFv3 process.

Use the **delete** form of this command to restore defaults for the router ID.

Use the **show** form of this command to display the router ID for an OSPFv3 process.



## protocols ospfv3 summary-address <address> <action>

Summarizes or suppresses external routes with a specified address range.

### Syntax

```
set protocols ospfv3 [ process process-id] summary-address { ipv6net | ipv4net } [ not-advertise | advertise tag tag-value ]
delete protocols ospfv3 [ process process-id] summary-address { ipv6net | ipv4net } [ not-advertise | advertise tag tag-value ]
show protocols ospfv3 [ process process-id] summary-address { ipv6net | ipv4net } [ not-advertise | advertise tag tag-value ]
```

### Command Default

Routing updates are not suppressed.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv6net*

A range of addresses given as IPv6 starting address and a mask.

*ipv4net*

A range of addresses given as IPv4 starting address and a mask.

**not-advertise**

Suppresses routes that match a range.

*tag-value*

Tag value to use as a "match" value for controlling redistribution via route maps. Set a tag value. The tag value ranges from 0 to 4294967295. The default is 0.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      summary-address
        ipv4net {
          not-advertise
          advertise tag tag-value
        }
        ipv6net {
          not-advertise
          advertise tag tag-value
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to summarize or suppress external routes with a specified address range.

An address range is a pairing of a starting address and a mask that is almost the same as IP network number. For example:

- If the specified IPV6 address range is 2020:100:100:2000::/53, it matches 2020:100:100:2222::/64, 2020:100:100:2666::/64 and so on.
- If the specified IPV4 address range is 192.168.0.0/255.255.240.0, it matches 192.168.1.0/24, 192.168.4.0/22, 192.168.8.128/25 and so on.

Redistributing routes from other protocols into OSPFv3 requires the router to advertise each route individually in an external LSA. Use this command to advertise one summary route for all redistributed routes covered by a specified network address and mask. This minimizes the size of the OSPFv3 link state database.

Use the **set** form of this command to summarize or suppress external routes with a specified address range.

Use the **delete** form of this command to remove summary addresses.

Use the **show** form of this command to display summary-addresses configuration.

## show ipv6 ospfv3

Displays high-level information about OSPFv3 configuration.

### Syntax

```
show ipv6 ospfv3 [process-id]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

### Modes

Operational mode

### Usage Guidelines

Use this command to display high-level information about OSPFv3 configuration.

## show ipv6 ospfv3 database

Displays OSPFv3 database information.

### Syntax

```
show ipv6 ospfv3 [process-id] database [adv-router | external | inter-prefix | inter-router | intra-prefix | link | max-age | network
| nssa-external | router | self-originate ] [ipv4] [adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

**external**

Displays information for the Autonomous System (AS) external LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database external](#) on page 119.

**inter-prefix**

Displays information for the Inter-Prefix LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database inter-prefix](#) on page 120.

**inter-router**

Displays information for the Inter-Router LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database inter-router](#) on page 121.

**intra-prefix**

Displays information for the Intra-Prefix LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database intra-prefix](#) on page 122.

**link**

Displays information for the Link LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database link](#) on page 123.

**max-age**

Displays information for the MaxAge LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database max-age](#) on page 124.

**network**

Displays information for the Network LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database network](#) on page 125.

**nssa-external**

Displays information for the not-so-stubby-area (NSSA) LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database nssa-external](#) on page 126.

**router**

Displays information for the Router LSAs within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database router](#) on page 127.

*ipv4*

Display the OSPFv3 database for a given address.

**adv-router** *ipv4*

Display the OSPFv3 database for a given address of the advertised router specified.

**self-originate**

Displays information for the self-originated link states within the OSPFv3 Link State database. For more information, see [show ipv6 ospfv3 database self-originate](#) on page 128.

## Modes

Operational mode

## Usage Guidelines

Use this command to display the OSPFv3 database information.

## show ipv6 ospfv3 database adv-router

Displays information for the advertising router within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [process-id] database adv-router ipv4
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Router ID of the advertising router. Specify an IP address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information relating to the advertising router in the OSPFv3 Link State database.

## show ipv6 ospfv3 database external

Displays information for the Autonomous System (AS) external LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [ process-id ] database external [ ipv4 [ [ adv-router ipv4 ] | [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

*adv-router ipv4*

Display the OSPFv3 database for a given address of the specified advertised router.

*self-originate*

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for the Autonomous System (AS) external LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database inter-prefix

Displays information for the Inter-Prefix LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [ process-id ] database inter-prefix [ ipv4 [ [ adv-router ipv4 ] ] [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

*adv-router ipv4*

Display the OSPFv3 database for a given address of the specified advertised router.

*self-originate*

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for Inter-Prefix LSAs within the OSPFv3 Link State database.



## show ipv6 ospfv3 database inter-router

Displays information for the Inter-Router LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [ process-id ] database inter-router [ ipv4 [ [ adv-router ipv4 ] | [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

*adv-router ipv4*

Display the OSPFv3 database for a given address of the specified advertised router.

*self-originate*

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for Inter-Router LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database intra-prefix

Displays information for the Intra-Prefix LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [ process-id ] database intra-prefix [ ipv4 [ [ adv-router ipv4 ] ] [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

*adv-router ipv4*

Display the OSPFv3 database for a given address of the specified advertised router.

*self-originate*

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for Intra-Prefix LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database link

Displays information for the Link LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [process-id] database link [ ipv4 [ [ adv-router ipv4 ] | [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

**adv-router *ipv4***

Display the OSPFv3 database for a given address of the specified advertised router.

**self-originate**

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for Link LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database max-age

Displays information for the MaxAge LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [process-id] database max-age
```

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for MaxAge LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database network

Displays information for the Network LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [ process-id ] database network [ ipv4 [ [ adv-router ipv4 ] ] [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

*adv-router ipv4*

Display the OSPFv3 database for a given address of the specified advertised router.

*self-originate*

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for Network LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database nssa-external

Displays information for the not-so-stubby-area (NSSA) LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [process-id] database nssa-external [ ipv4 [ [ adv-router ipv4 ] ] [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

*adv-router ipv4*

Display the OSPFv3 database for a given address of the specified advertised router.

*self-originate*

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for NSSA LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database router

Displays information for the router LSAs within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [process-id] database router [ ipv4 [ [ adv-router ipv4 ] | [ self-originate ] ] | adv-router ipv4 | self-originate ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

The Link State ID. Display the OSPFv3 database for a given address.

**adv-router *ipv4***

Display the OSPFv3 database for a given address of the specified advertised router.

**self-originate**

Display the self-originate OSPFv3 database for a given address.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for Router LSAs within the OSPFv3 Link State database.

## show ipv6 ospfv3 database self-originate

Displays information for the self-originated link states within the OSPFv3 Link State database.

### Syntax

```
show ipv6 ospfv3 [process-id] database self-originate
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information for the self-originated link states within the OSPFv3 Link State database.



## show ipv6 ospfv3 interface

Displays OSPFv3 configuration and status information for a specified interface.

### Syntax

```
show ipv6 ospfv3 interface [ interface ]
```

### Command Default

If no interfaces are specified then information on all interfaces is displayed.

### Parameters

*interface*

The identifier of an interface. Supported interface types are:

- Data plane
- Loopback

For more information about these interface types, refer to [Supported Interface Types](#) on page 181.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information about OSPFv3 configuration for an interface.

## show ipv6 ospfv3 neighbor

Displays OSPFv3 neighbor information for a specified address or interface.

### Syntax

```
show ipv6 ospfv3 [process-id] neighbor [ ipv4 | detail | /| interface ]
```

### Command Default

If no interfaces are specified, then information on all neighbors is displayed.

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*ipv4*

Display neighbor information for the specified address.

**detail**

Display detailed neighbor information for all neighbors.

*interface*

The identifier of an interface. Supported interface types are:

- Data plane
- Loopback

For more information about these interface types, refer to [Supported Interface Types](#) on page 181.

### Modes

Operational mode

### Usage Guidelines

Use this command to display OSPFv3 neighbor information for a specified address or interface.

## show ipv6 ospfv3 route

Displays OSPFv3 route information.

### Syntax

```
show ipv6 ospfv3 [process-id] route
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

### Modes

Operational mode

### Usage Guidelines

Use this command to display the OSPFv3 route information.

## show ipv6 ospfv3 topology

Displays information about OSPFv3 topology for all areas or a specified area.

### Syntax

```
show ipv6 ospfv3 [process-id] topology [ area area-id ]
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967295.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information about OSPFv3 topology for all areas or a specified area.

## show ipv6 ospfv3 virtual-links

Displays information about OSPFv3 virtual links.

### Syntax

```
show ipv6 ospfv3 [process-id] virtual-links
```

### Parameters

*process-id*

An OSPFv3 process ID. Enter alphanumeric characters.

### Modes

Operational mode

### Usage Guidelines

Use this command to display information about OSPFv3 virtual links.

## show ipv6 route ospfv3

Displays all IPv6 OSPFv3 routes.

### Syntax

```
show ipv6 route ospfv3
```

### Modes

Operational mode

### Usage Guidelines

Use this command to display all the IPv6 OSPFv3 routes.

## show monitoring protocols ospfv3

Displays OSPFv3 protocol debugging flags.

### Syntax

```
show monitoring protocols ospfv3
```

### Modes

Operational mode

### Usage Guidelines

Use this command to display OSPFv3 protocol debugging flags. This command allows you to see how debugging is set for OSPFv3.





# OSPFv3 Area Commands

---

- protocols ospfv3 area <area-id>..... 138
- protocols ospfv3 area <area-id> nssa..... 139
- protocols ospfv3 area <area-id> nssa default-cost..... 140
- protocols ospfv3 area <area-id> nssa default-information-originate..... 141
- protocols ospfv3 area <area-id> nssa default-information-originate metric..... 142
- protocols ospfv3 area <area-id> nssa default-information-originate metric-type..... 143
- protocols ospfv3 area <area-id> nssa no-redistribution..... 144
- protocols ospfv3 area <area-id> nssa no-summary..... 145
- protocols ospfv3 area <area-id> nssa stability-interval..... 146
- protocols ospfv3 area <area-id> nssa translator-role..... 147
- protocols ospfv3 area <area-id> range..... 148
- protocols ospfv3 area <area-id> stub..... 150
- protocols ospfv3 area <area-id> virtual-link <router-id>..... 152
- protocols ospfv3 area <area-id> virtual-link <router-id> dead-interval <interval>..... 153
- protocols ospfv3 area <area-id> virtual-link <router-id> hello-interval <interval>..... 155
- protocols ospfv3 area <area-id> virtual-link <router-id> retransmit-interval <interval>..... 157
- protocols ospfv3 area <area-id> virtual-link <router-id> transmit-delay <delay>..... 159

## protocols ospfv3 area <area-id>

Defines an OSPFv3 area within an Autonomous System (AS).

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id
delete protocols ospfv3 [ process process-id] area area-id
show protocols ospfv3 [ process process-id] area area-id
```

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id
    }
  }
}
```

### Usage Guidelines

Use this command to define an area within an OSPFv3 AS.

Use the **set** form of this command to create an OSPFv3 area or define the parameters for a specified OSPFv3 area.

Use the **delete** form of this command to remove an OSPFv3 area.

Use the **show** form of this command to display the OSPFv3 area configuration.

## protocols ospfv3 area <area-id> nssa

Designates an OSPFv3 area as a not-so-stubby area (NSSA).

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id nssa
delete protocols ospfv3 [ process process-id] area area-id nssa
show protocols ospfv3 [ process process-id] area area-id nssa
```

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa
      }
    }
  }
}
```

### Usage Guidelines

Use this command to designate an OSPFv3 area as an NSSA. There are no external routes in an OSPFv3 stub area. Therefore you cannot redistribute from another protocol into a stub area. An NSSA allows external routes to be flooded within the area. These routes are then leaked into other areas. However, the external routes from other areas still do not enter the NSSA. You can configure an area to be a stub area or an NSSA, but not both.

This command simplifies administration when connecting a central site that is using OSPFv3 to a remote site that is using a different routing protocol. You can extend OSPFv3 to cover the remote connection by defining the area between the central router and the remote router as an NSSA.

Type 5 AS-external LSAs are not allowed in stubby areas. Type 7 LSAs may be translated into Type 5 LSAs by the NSSA Area Border Router (ABR) and may traverse the NSSA in this manner. Inter-area routes are not allowed.

Use the **set** form of this command to designate an OSPFv3 area as an NSSA.

Use the **delete** form of this command to remove an NSSA designation.

Use the **show** form of this command to display an NSSA configuration.

## protocols ospfv3 area <area-id> nssa default-cost

Sets the administrative cost, or metric, to be applied to the OSPFv3 not-so-stubby-area (NSSA).

### Syntax

**set** protocols ospfv3 [ process *process-id*] area *area-id* nssa default-cost *cost*

**delete** protocols ospfv3 [ process *process-id*] area *area-id* nssa default-cost

**show** protocols ospfv3 [ process *process-id*] area *area-id* nssa default-cost

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

*cost*

The administrative cost, or metric, that is applied to an NSSA. The range is 0 to 6777215. The default is 1.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          default-cost cost
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the administrative cost or metric to be applied to the OSPFv3 NSSA.

Use the **set** form of this command to set the administrative cost or metric to be applied to the OSPFv3 NSSA.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> nssa default-information-originate

Sets a default external route into an OSPFv3 routing area.

### Syntax

**set** protocols ospfv3 [ *process process-id* ] **area** *area-id* **nssa default-information-originate**

**delete** protocols ospfv3 [ *process process-id* ] **area** *area-id* **nssa default-information-originate**

**show** protocols ospfv3 [ *process process-id* ] **area** *area-id* **nssa default-information-originate**

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          default-information-originate
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set a default external route into an OSPFv3 routing area.

Use the **set** form of this command to set a default external route into an OSPFv3 routing area.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> nssa default-information-originate metric

Sets an OSPFv3 default metric into an OSPFv3 routing area.

### Syntax

**set** protocols ospfv3 [ process *process-id*] area *area-id* nssa default-information-originate metric

**delete** protocols ospfv3 [ process *process-id*] area *area-id* nssa default-information-originate metric

**show** protocols ospfv3 [ process *process-id*] area *area-id* nssa default-information-originate metric

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          default-information-originate metric
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set an OSPFv3 default metric into an OSPFv3 routing area.

Use the **set** form of this command to set an OSPFv3 default metric into an OSPFv3 routing area.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> nssa default-information-originate metric-type

Sets an OSPFv3 metric type for default routes.

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id nssa default-information-originate metric-type
delete protocols ospfv3 [ process process-id] area area-id nssa default-information-originate metric-type
show protocols ospfv3 [ process process-id] area area-id nssa default-information-originate metric-type
```

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          default-information-originate metric-type
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set an OSPFv3 metric type for default routes.

Use the **set** form of this command to set OSPFv3 metric type for default routes.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> nssa no-redistribution

Configures a not-so-stubby-area (NSSA) router to disallow redistributed link-state advertisement (LSA) from entering an NSSA.

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id nssa no-redistribution
delete protocols ospfv3 [ process process-id] area area-id nssa no-redistribution
show protocols ospfv3 [ process process-id] area area-id nssa no-redistribution
```

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          no-redistribution
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to configure an NSSA router to disallow redistributed LSAs from entering the NSSA. Use this command when the router is both an NSSA Autonomous System Boundary Router (ASBR) and an NSSA Area Border Router (ABR).

Use the **set** form of this command to configure an NSSA router to disallow redistributed LSAs from entering the NSSA.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.



## protocols ospfv3 area <area-id> nssa no-summary

Configures OSPFv3 so that inter-area routes are not injected into a not-so-stubby-area (NSSA).

### Syntax

**set** protocols ospfv3 [ process *process-id*] area *area-id* nssa no-summary

**delete** protocols ospfv3 [ process *process-id*] area *area-id* nssa no-summary

**show** protocols ospfv3 [ process *process-id*] area *area-id* nssa no-summary

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          no-summary
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to configure OSPFv3 so that inter-area routes are not injected into an NSSA. This command prevents route summaries from being generated into the area.

Use the **set** form of this command to configure OSPFv3 so that inter-area routes are not injected into an NSSA.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> nssa stability-interval

Configures a stability timer for a not-so-stubby-area (NSSA).

### Syntax

**set** protocols ospfv3 [ process *process-id*] area *area-id* nssa stability-interval *interval*

**delete** protocols ospfv3 [ process *process-id*] area *area-id* nssa stability-interval

**show** protocols ospfv3 [ process *process-id*] area *area-id* nssa stability-interval

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

*interval*

Stability interval in seconds. The range is from 0 to 2147483647.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          stability-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to configure a stability timer for an NSSA. If an elected translator determines its services are no longer required, it continues to perform its duties for this time interval. This minimizes excess flushing of translated Type-7 LSAs and provides a more stable translator transition.

Use the **set** form of this command to configure a stability timer for an NSSA.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> nssa translator-role

Specifies the translator role of the not-so-stubby-area (NSSA) Area Border Router (ABR) router in a NSSA.

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id nssa translator-role { always | candidate }
delete protocols ospfv3 [ process process-id] area area-id nssa translator-role { always | candidate }
show protocols ospfv3 [ process process-id] area area-id nssa translator-role
```

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

**always**

The NSSA-ABR translator role of always. This role allows the NSSA-ABR router to translate Type 7 link-state advertisements (LSAs) into Type 5 LSAs.

**candidate**

The NSSA-ABR translator role of candidate. This role allows the NSSA-ABR router to translate Type 7 LSAs into Type 5 LSAs, if a router is elected.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        nssa {
          translator-role always
          translator-role candidate
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify the translator role of an NSSA-ABR router in an NSSA.

Use the **set** form of this command to specify the translator role of an NSSA-ABR router in an NSSA.

Use the **delete** form of this command to remove the area type configuration.

Use the **show** form of this command to display the area type configuration.

## protocols ospfv3 area <area-id> range

Configures an OSPFv3 address range.

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id range {ipv6net} [ advertise | not-advertise ]
delete protocols ospfv3 [ process process-id] area area-id range {ipv6net} [ advertise | not-advertise ]
show protocols ospfv3 [ process process-id] area area-id range {ipv6net} [ advertise | not-advertise ]
```

### Command Default

By default, routes are advertised and routes are not substituted.

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value. The decimal value ranges from 0 to 4294967.

*ipv6net*

The range to be summarized, expressed as an IPv6 network in the format *ipv6-address/prefix*.

**advertise**

Advertise this range.

**not-advertise**

Do not advertise this range. Directs the router not to advertise routes in this range.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        range {
          ipv6net {
            advertise
            not-advertise
          }
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to configure an OSPFv3 address range. This command summarizes intra-area routes for an Area Border Router (ABR). Use this command only with an ABR.

The **range** parameter cannot co-exist with NSSA stub or virtual-link configuration if configured in the same area. This means the **range** parameter can be used only if the area type is normal (or unspecified, because normal is the default type), not stub or NSSA. In addition, the **range** parameter cannot be used within an area that is configured as a virtual link.

The single summary route is then advertised to other areas by the ABRs. Routing information is condensed at area boundaries and outside the area. If the network numbers in an area are assigned in a way such that they are contiguous, the ABRs can be configured to advertise a summary route that covers all the individual networks within the area that fall into the specified range.

Use the **set** form of this command to direct the router to summarize routes matching a prefix range.

Use the **delete** form of this command to remove the assigned area range.

Use the **show** form of this command to display area-range configuration.

## protocols ospfv3 area <area-id> stub

Designates an OSPFv3 area as a stub area.

### Syntax

```
set protocols ospfv3 [ process process-id ] area area-id stub [ default-cost cost | no-summary ]
delete protocols ospfv3 [ process process-id ] area area-id stub [ default-cost | no-summary ]
show protocols ospfv3 [ process process-id ] area area-id stub [ default-cost | no-summary ]
```

### Command Default

By default, summary routes are generated into the area.

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value.

*cost*

The administrative cost, or metric, that is applied to the default route in this area. The range is 0 to 6777215.

**no-summary**

Prevents route summaries from being generated into the area.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        stub {
          default-cost cost
          no-summary
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to designate an OSPFv3 area as a stub area. No Type 5 Autonomous System (AS)-external link-state advertisement (LSAs) are allowed into a stub area.

Use the **set** form of this command to set the OSPFv3 area type to a stub area.

Use the **delete** form of this command to remove area type configuration.

Use the **show** form of this command to display an area type configuration.

## protocols ospfv3 area <area-id> virtual-link <router-id>

Configures a link between two backbone areas that are physically separated through other non-backbone areas.

### Syntax

**set** protocols ospfv3 [ **process** *process-id* ] **area** *area-id* **virtual-link** *router-id*

**delete** protocols ospfv3 [ **process** *process-id* ] **area** *area-id* **virtual-link** *router-id*

**show** protocols ospfv3 [ **process** *process-id* ] **area** *area-id* **virtual-link** *router-id*

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value.

*router-id*

The router ID for an OSPFv3 process. The format is an IPv4 address.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        virtual-link router-id {
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to configure a link between two backbone areas that are physically separated through other non-backbone areas.

In OSPFv3, all non-backbone areas must be connected to a backbone area. If the connection to the backbone is lost, the virtual link repairs the connection. You can configure virtual links between any two backbone routers that have an interface to a common non-backbone area. The protocol treats these two routers joined by a virtual link as if they were connected by an unnumbered point-to-point network.

Use the **set** form of this command to configure a link between two backbone areas that are physically separated through other non-backbone areas.

Use the **delete** form of this command to remove the virtual link.

Use the **show** form of this command to display virtual-link configuration.



## protocols ospfv3 area <area-id> virtual-link <router-id> dead-interval <interval>

Specifies the dead interval for a virtual link.

### Syntax

**set protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **dead-interval** *interval*

**delete protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **dead-interval**

**show protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **dead-interval**

### Command Default

The dead interval is 40 seconds.

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value.

*router-id*

The router ID for an OSPFv3 process. The format is an IPv4 address.

*interval*

An interval in seconds during which the virtual link should wait to detect Hello packets from neighboring routers before declaring the neighbor down. The range is from 1 to 65535. The default is 40 seconds.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        virtual-link router-id {
          dead-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify the interval during which a virtual link should expect a hello packet from its neighbor.

If the dead interval passes without the interface receiving a hello packet from the neighbor, the status of the neighbor is changed to out-of-service, and all associated state is cleared.

The dead interval must be the same for all routers that are to establish two-way communication within a network. If two routers do not agree on these parameters, they do not establish adjacencies and disregard communication from each other.

#### NOTE

Ensure that you do not set a dead interval that is too short in seconds. Otherwise, the configuration may end up declaring the virtual link prematurely down and may end up partitioning the backbone.

Use the **set** form of this command to specify the dead interval for a virtual link.

Use the **delete** form of this command to restore a deal interval of 40 seconds.

Use the **show** form of this command to display dead-interval configuration.

## protocols ospfv3 area <area-id> virtual-link <router-id> hello-interval <interval>

Sets the interval between OSPFv3 Hello packets on a virtual link.

### Syntax

**set protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **hello-interval** *interval*

**delete protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **hello-interval**

**show protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **hello-interval**

### Command Default

Hello packets are sent every 10 seconds.

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value.

*router-id*

The router ID for an OSPFv3 process. The format is an IPv4 address.

*interval*

The interval, in seconds, between Hello packets. This value must be the same for all nodes on the network. The range is 1 to 65535. The default is 10.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        virtual-link router-id {
          hello-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the interval at which OSPFv3 Hello packets are sent for a virtual link.

A hello packet is an OSPFv3 packet used to detect and maintain relationships with neighbors on the same network (directly connected routers). The greater the interval between Hello packets, the less router traffic occurs, but the longer it takes for topological changes to be detected.

The hello interval must be the same for all routers that are to establish two-way communication within a network. If two routers do not agree on these parameters, they do not establish adjacencies and disregard communication from each other.

Use the **set** form of this command to set the hello interval.

Use the **delete** form of this command to restore a hello interval of 10 seconds.

Use the **show** form of this command to display hello-interval configuration.

## protocols ospfv3 area <area-id> virtual-link <router-id> retransmit-interval <interval>

Specifies the retransmit interval for a virtual link.

### Syntax

```
set protocols ospfv3 [ process process-id] area area-id virtual-link router-id retransmit-interval interval
delete protocols ospfv3 [ process process-id] area area-id virtual-link router-id retransmit-interval
show protocols ospfv3 [ process process-id] area area-id virtual-link router-id retransmit-interval
```

### Command Default

Unacknowledged LSAs are retransmitted at five-second intervals.

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value.

*router-id*

The router ID for an OSPFv3 process. The format is an IPv4 address.

*interval*

The interval, in seconds, between retransmitting unacknowledged link-state advertisements (LSAs). This interval must be the same for all nodes on the network. The range is 1 to 65535. The default interval is 5.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        virtual-link router-id {
          retransmit-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the retransmit interval for a virtual link. The interval is the number of seconds before retransmitting an unacknowledged LSA.

When an OSPFv3 router sends an LSA to a neighbor, the neighbor acknowledges receipt with a link-state acknowledgment (LS Ack) packet. If the local router fails to receive the expected LS Ack packet, it retransmits the LSA at the interval specified by this command. This interval must be the same for all nodes on the network.

Use the **set** form of this command to set a retransmit interval for a virtual link.

Use the **delete** form of this command to restore a retransmit interval of five seconds for unacknowledged LSAs.

Use the **show** form of this command to display retransmit-interval configuration.

## protocols ospfv3 area <area-id> virtual-link <router-id> transmit-delay <delay>

Specifies the transmit delay for a virtual link.

### Syntax

**set protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **transmit-delay** *delay*

**delete protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **transmit-delay**

**show protocols ospfv3** [**process** *process-id*] **area** *area-id* **virtual-link** *router-id* **transmit-delay**

### Command Default

Link-state transmits occur at one-second intervals.

### Parameters

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*area-id*

The ID of an OSPFv3 area being configured, expressed as an IP address or a decimal value.

*router-id*

The router ID for an OSPFv3 process. The format is an IPv4 address.

*delay*

The delay, in seconds, between link state transmits. This value must be the same for all nodes on the network. The range is 1 to 65535. The default is 1.

### Modes

Configuration mode

### Configuration Statement

```
protocols {
  ospfv3 {
    process process-id {
      area area-id {
        virtual-link router-id {
          transmit-delay delay
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the transmit delay for a virtual link in an area. This delay is the estimated time required to send a link-state update (LSU) packet.

This timer is used to accommodate transmission and propagation delays on the network, particularly on low-speed networks in which delays may be significant. The router increments the age of link-state advertisements (LSAs) in LSU packets to account for these delays.

The delay includes both the transmission time and the propagation delay across the network. The transmit delay is added to the age of the LSA packet before the LSA is transmitted. The LSA age is used to help the network sequence LSAs, so that it can determine which of competing LSAs is the more recent and trustworthy.

LSAs are numbered in sequence, but the sequence numbers are finite, and so cannot be used as the sole determinant of the most recent LSA. Instead, OSPFv3 also tracks the age of LSAs. Each time the LSA is forwarded to another router, its current age is incremented by the transmit delay. The age of the packet, together with its sequence number, helps the receiving router to determine which version of a received LSA is more recent, and therefore to be used.

Use the **set** form of this command to set the transmit delay. Set the transmit-delay to be greater than zero. Also, take into account the transmission and propagation delays for the interface.

Use the **delete** form of this command to restore an interval of one second for link-state transmits.

Use the **show** form of this command to display the transmit-delay configuration.



# OSPFv3 Interface Commands

---

- interfaces <interface> ipv6 ospfv3.....162
- interfaces <interface> ipv6 ospfv3 cost <cost>.....163
- interfaces <interface> ipv6 ospfv3 dead-interval <interval>.....165
- interfaces <interface> ipv6 ospfv3 hello-interval <interval>.....167
- interfaces <interface> ipv6 ospfv3 link-lsa-suppression enable.....169
- interfaces <interface> ipv6 ospfv3 mtu-ignore.....170
- interfaces <interface> ipv6 ospfv3 neighbor.....172
- interfaces <interface> ipv6 ospfv3 network <type>.....173
- interfaces <interface> ipv6 ospfv3 priority <priority>.....175
- interfaces <interface> ipv6 ospfv3 retransmit-interval <interval>.....177
- interfaces <interface> ipv6 ospfv3 transmit-delay <delay>.....179

## interfaces <interface> ipv6 ospfv3

Enables OSPFv3 on a specified interface.

### Syntax

**set interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ]

**delete interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ]

**show interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ]

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, refer to [Supported Interface Types](#) on page 181.

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to enable the OSPFv3 routing protocol on an interface.

Use the **set** form of this command to enable OSPFv3 on an interface.

Use the **delete** form of this command to remove all OSPFv3 configuration and disable OSPFv3 on an interface.

Use the **show** form of this command to display OSPFv3 configuration.

## interfaces <interface> ipv6 ospfv3 cost <cost>

Sets the routing cost for OSPFv3 on an interface.

### Syntax

**set interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ] **cost** *cost*

**delete interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ] **cost**

**show interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ] **cost**

### Command Default

For details on the default cost of OSPFv3, see the Usage Guidelines.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 cost <cost>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*cost*

The link-state metric (OSPFv3 cost) to be advertised in the link-state advertisement (LSA) as the cost of sending packets over the interface. The range is 1 to 65535.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          cost cost
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to manually override the default OSPFv3 cost computed by the system for an interface. You can assign only one cost for each interface.

By default, the metric associated with a link is computed as follows:

$$\text{Cost} = 108/\text{bandwidth}$$

The cost of reaching any destination is the sum of the costs of the individual hops. Costs are always rounded to the nearest integer. Costs lower than 1 are rounded up to 1.

Table 3 shows the OSPFv3 costs for some common media types.

Media Type	OSPF Cost
56 Kbps	1785
64 Kbps	1562
128 Kbps	781
256 Kbps	390
512 Kbps	195
768 Kbps	130
T1 (1.544 Mbps)	64
E1 (2.048 Mbps)	48
4 Mbps Token Ring	6
10 Mbps Ethernet	10
16 Mbps Token Ring	6
T3 (44.736 Mbps)	2
100+ Mbps	1

The values in Table 3 show how OSPFv3 fails to distinguish between interfaces faster than 100 Mbps, for example, between Fast Ethernet (100 Mbps) and Gigabit Ethernet (1000 Mbps) interfaces. If you want to distinguish interfaces equal to or greater than 100 Mbps, you must manually configure the cost of the interface by using this command.

Use the **set** form of this command to specify the OSPFv3 cost for the interface.

Use the **delete** form of this command to restore the default cost.

Use the **show** form of this command to display cost configuration.

## interfaces <interface> ipv6 ospfv3 dead-interval <interval>

Sets the OSPFv3 dead interval for an interface.

### Syntax

```
set interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id] instance-id dead-interval interval
delete interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id] dead-interval
show interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id] dead-interval
```

### Command Default

The default for the dead interval is 40 seconds.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 dead-interval <interval>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*interval*

The time, in seconds, that this interface should wait to detect Hello packets from neighboring routers before declaring the neighbor down. The range is 1 to 65535. The default is 40 seconds.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          dead-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify the interval during which an interface should expect a hello packet from its neighbor.

If the dead interval passes without the interface receiving a hello packet from the neighbor, the status of the neighbor is changed to out-of-service, and all associated state is cleared.

The dead interval must be the same for all routers that are to establish two-way communication within a network. If two routers do not agree on these parameters, they do not establish adjacencies and disregard communication from each other.

Use the **set** form of this command to specify the dead interval.

Use the **delete** form of this command to restore the default dead interval.

Use the **show** form of this command to display dead-interval configuration.

## interfaces <interface> ipv6 ospfv3 hello-interval <interval>

Sets the interval between OSPFv3 Hello packets on an interface.

### Syntax

```
set interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] hello-interval interval
delete interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] hello-interval
show interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] hello-interval
```

### Command Default

Hello packets are sent every 10 seconds.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 hello-interval <interval>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*interval*

The interval, in seconds, between Hello packets. This value must be the same for all nodes on the network. The range is 1 to 65535. The default is 10.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          hello-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the interval at which OSPFv3 Hello packets are sent for an interface.

A hello packet is an OSPFv3 packet used to detect and maintain relationships with neighbors on the same network (directly connected routers). The greater the interval between Hello packets, the less router traffic occurs, but the longer it takes for topological changes to be detected.

The hello interval must be the same for all routers that are to establish two-way communication within a network. If two routers do not agree on these parameters, they do not establish adjacencies and disregard communication from each other.

Use the **set** form of this command to set the hello interval.

Use the **delete** form of this command to restore the default hello interval.

Use the **show** form of this command to display hello-interval configuration.



## interfaces <interface> ipv6 ospfv3 link-lsa-suppression enable

Enables link-lsa-suppression for an instance identifier on an interface.

### Syntax

**set interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] link-lsa-suppression enable

**delete interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] link-lsa-suppression

**show interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] link-lsa-suppression

### Command Default

By default, the suppression of the origination of the link-LSAs is disabled.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 link-lsa-suppression enable](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          link-lsa-suppression enable
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to suppress the origination of the link-LSAs. If this feature is enabled, and the interface type is not broadcast or Non-Broadcast Multi-Access (NBMA), the router does not originate a link-LSA for the interface.

Use the **set** form of this command to suppress the origination of the link-LSAs.

Use the **delete** form of this command to restore the default setting.

Use the **show** form of this command to display whether link-LSA suppression is enabled.

## interfaces <interface> ipv6 ospfv3 mtu-ignore

Disables maximum transmission unit (MTU) mismatch detection on an OSPFv3 interface.

### Syntax

**set interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ] **mtu-ignore**

**delete interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ] **mtu-ignore**

**show interfaces** *interface* **ipv6 ospfv3** [ **process** *process-id* **instance-id** *instance-id* ]

### Command Default

MTU mismatch detection is enabled by default.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 mtu-ignore](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          mtu-ignore
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to disable MTU mismatch detection on an OSPFv3 interface.

OSPFv3 sends the MTU of the interface in a database description packet. If the MTUs of OSPFv3 neighbors do not match, they cannot form an OSPFv3 adjacency. MTU mismatch detection detects MTU mismatches and indicates them in the form of a debug message.

MTU mismatch is an important troubleshooting feature. If MTU mismatch is not enabled, MTU mismatches can be detected only by examining the configuration for both interfaces.

MTU mismatches are unavoidable in some network setups and are even part of the normal setup. MTU mismatch detection should be disabled for these networks, so that normal OSPFv3 adjacencies can be formed.

Use the **set** form of this command to disable MTU mismatch detection.

Use the **delete** form of this command to re-enable MTU mismatch detection.

Use the **show** form of this command to display the OSPFv3 configuration.

## interfaces <interface> ipv6 ospfv3 neighbor

Configures an IPv6 address for neighbor parameters on an interface.

### Syntax

**set** interfaces *interface* ipv6 ospfv3 [ process *process-id* instance-id *instance-id*] neighbor

**delete** interfaces *interface* ipv6 ospfv3 [ process *process-id* instance-id *instance-id*] neighbor

**show** interfaces *interface* ipv6 ospfv3 [ process *process-id* instance-id *instance-id*]

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 neighbor](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          neighbor
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to configure an IPv6 address for neighbor parameters on an interface.

Use the **set** form of this command to configure an IPv6 address for neighbor parameters.

Use the **delete** form of this command to remove the neighbor parameters.

Use the **show** form of this command to display the OSPFv3 configuration.

## interfaces <interface> ipv6 ospfv3 network <type>

Specifies the OSPFv3 network type for an interface.

### Syntax

**set interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] **network** [ **broadcast** | **non-broadcast** | **point-to-multipoint** | **point-to-point** ]

**delete interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] **network**

**show interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] **network**

### Command Default

The default network type is broadcast.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 network <type>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*type*

The OSPFv3 network type of the interface. Supported values are as follows:

**broadcast:** The interface supports broadcast mode, such as a LAN link.

**non-broadcast:** The interface does not support broadcast mode.

**point-to-multipoint:** The interface supports point-to-multipoint mode, such as a Point-to-Point Protocol (PPP) interface or a point-to-point logical interface on Frame Relay.

**point-to-point:** The interface supports point-to-point mode, such as a Non-Broadcast Multi-Access (NBMA) interface. The default is broadcast.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          network type
        }
      }
    }
  }
}
```

## Usage Guidelines

Use this command to configure and display the OSPFv3 network type for an interface.

Use the **set** form of this command to specify the network type.

Use the **delete** form of this command to remove the network type.

Use the **show** form of this command to display the network type.

## interfaces <interface> ipv6 ospfv3 priority <priority>

Sets the OSPFv3 priority for an interface.

### Syntax

```
set interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] priority priority
delete interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] priority
show interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] priority
```

### Command Default

An OSPFv3 interface has a priority of 1.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 priority <priority>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*priority*

An OSPFv3 router priority for the interface. The range is 0 to 255, for which a router with priority 0 can never become the designated router. The default is 1.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          priority priority
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the priority for an interface on the broadcast network to which the interface is connected. The priority determines which routers are selected as the Designated Router (DR) and Backup Designated Router (BDR) of the area.

The DR and BDR are used to reduce the amount of traffic in OSPFv3 overhead on broadcast networks, by reducing the number of adjacent routers to which a router must flood its topological information. In broadcast networks (such as Ethernet), each router establishes an adjacency with only the DR and the BDR, rather than with every router in its area. The DR and the BDR then flood this information to all other routers on the network segment.

Priority can range from 0 to 255. In general, the router with the highest priority is elected as the DR, and the router with the second-highest priority is elected as the BDR. The higher the number, the higher the priority.

Routers with a priority of 0 are ineligible for election.

Use the **set** form of this command to specify the OSPFv3 priority.

Use the **delete** form of this command to restore the default priority.

Use the **show** form of this command to display priority configuration.



## interfaces <interface> ipv6 ospfv3 retransmit-interval <interval>

Sets the OSPFv3 retransmit interval for an interface.

### Syntax

**set interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] **retransmit-interval** *interval*

**delete interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] **retransmit-interval**

**show interfaces** *interface* ipv6 ospfv3 [ **process** *process-id* **instance-id** *instance-id* ] **retransmit-interval**

### Command Default

Unacknowledged LSAs are retransmitted at five-second intervals.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 retransmit-interval <interval>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*interval*

The time, in seconds, to wait for an acknowledgement, after which the system retransmits a link-state advertisement (LSA) packet to its neighbors. The range is 3 to 65535. The default is 5.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          retransmit-interval interval
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to specify how long an interface waits for an acknowledgment of a link-state update before resending the update.

The link-state update packet is part of the exchange of topology databases between routers. When the initial database description (DD) packet is sent, it contains only the headers of the LSAs. If the receiving router determines that it requires that piece of the OSPFv3 topology, it sends a link-state request packet to request the complete LSA from the sending router.

After the update packet is sent, the sending router waits for an acknowledgement, either implicit or explicit, from the receiving router. In an explicit acknowledgement, the receiving router sends a link-state acknowledge (LS-Ack) packet to the router that sent the update. In an implicit acknowledgement, the router that sent the update receives an LSA from the receiving router that contains the update information.

If the retransmit interval passes with neither an explicit nor an implicit acknowledgement, the sending router retransmits the link-state update packet.

Too high an interval slows network convergence. Too small an interval causes unnecessary retransmission.

Use the **set** form of this command to set the OSPFv3 retransmit interval for an interface.

Use the **delete** form of this command to restore the default retransmit interval.

Use the **show** form of this command to display retransmit interval configuration.

## interfaces <interface> ipv6 ospfv3 transmit-delay <delay>

Specifies the OSPFv3 transmit delay for an interface.

### Syntax

```
set interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] transmit-delay delay
delete interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] transmit-delay
show interfaces interface ipv6 ospfv3 [ process process-id instance-id instance-id ] transmit-delay
```

### Command Default

Link-state transmits occur at one-second intervals.

### Parameters

*interface*

The type of interface. For detailed keywords and arguments that can be specified as interface types, see [interfaces <interface> ipv6 ospfv3 transmit-delay <delay>](#).

*process-id*

The OSPFv3 process ID. Enter alphanumeric characters.

*instance-id*

The OSPFv3 instance ID. The range is from 0 to 255.

*delay*

The delay, in seconds, between link-state transmits. This value must be the same for all nodes on the network. The range is 1 to 65535. The default is 1.

### Modes

Configuration mode

### Configuration Statement

```
interfaces interface {
  ipv6 {
    ospfv3 {
      process process-id {
        instance-id instance-id {
          transmit-delay delay
        }
      }
    }
  }
}
```

### Usage Guidelines

Use this command to set the transmit delay for an interface. This delay is the estimated time required to send a link-state update (LSU) packet.

This timer is used to accommodate transmission and propagation delays on the network, particularly on low-speed networks in which delays may be significant. The router increments the age of link-state advertisements (LSA) in LSU packets to account for these delays.

The value includes both the transmission time and the propagation delay across the network. The transmit delay is added to the age of the LSA packet before the LSA is transmitted. The LSA age is used to help the network sequence LSAs, so that it can determine which of competing LSAs is the more recent and trustworthy.

LSAs are numbered in sequence, but the sequence numbers are finite, and so cannot be used as the sole determinant of the most recent LSA. Instead, OSPFv3 also tracks the age of LSAs. Each time an LSA is forwarded to another router, its current age is incremented by the transmit delay. The age of a packet, together with its sequence number, helps the receiving router to determine which version of a received LSA is more recent, and therefore to be used.

Use the **set** form of this command to set the transmit delay.

Use the **delete** form of this command to restore the default transmit delay.

Use the **show** form of this command to display transmit-delay configuration.

# Supported Interface Types

The following table shows the syntax and parameters of supported interface types. Depending on the command, some of these types may not apply.

Interface Type	Syntax	Parameters
Bridge	<b>bridge</b> <i>brx</i>	<i>brx</i> : The name of a bridge group. The name ranges from br0 through br999.
Data plane	<b>dataplane</b> <i>interface-name</i>	<p><i>interface-name</i>: The name of a data plane interface. Following are the supported formats of the interface name:</p> <ul style="list-style-type: none"> <li>• <b>dp<math>x</math>py<math>z</math></b>—The name of a data plane interface, where <ul style="list-style-type: none"> <li>— <b>dp<math>x</math></b> specifies the data plane identifier (ID). Currently, only dp0 is supported.</li> <li>— <b>py</b> specifies a physical or virtual PCI slot index (for example, p129).</li> <li>— <b>p<math>z</math></b> specifies a port index (for example, p1). For example, dp0p1p2, dp0p160p1, and dp0p192p1.</li> </ul> </li> <li>• <b>dp<math>x</math>em<math>y</math></b>—The name of a data plane interface on a LAN-on-motherboard (LOM) device that does not have a PCI slot, where <b>em<math>y</math></b> specifies an embedded network interface number (typically, a small number). For example, dp0em3.</li> <li>• <b>dp<math>x</math>s<math>y</math></b>—The name of a data plane interface on a device that is installed on a virtual PCI slot, where <b>s<math>y</math></b> specifies an embedded network interface number (typically, a small number). For example, dp0s2. Currently, this format applies only when using the KVM or Hyper-V platforms.</li> <li>• <b>dp<math>x</math>P<math>n</math>py<math>z</math></b>—The name of a data plane interface on a device that is installed on a secondary PCI bus, where <b>P<math>n</math></b> specifies the bus number. You can use this format to name data plane interfaces on large physical devices with multiple PCI buses. For these devices, it is possible to have network interface cards installed on different buses with these cards having the same slot ID. The value of <i>n</i> must be an integer greater than 0. For example, dp0P1p162p1 and dp0P2p162p1.</li> </ul>
Data plane vif	<b>dataplane</b> <i>interface-name</i> <b>vif</b> <i>vif-id</i> [ <b>vlan</b> <i>vlan-id</i> ]	<p><i>interface-name</i>: Refer to the preceding description.</p> <p><i>vif-id</i>: A virtual interface ID. The ID ranges from 1 through 4094.</p> <p><i>vlan-id</i>: The VLAN ID of a virtual interface. The ID ranges from 1 through 4094.</p>
Loopback	<b>loopback</b> <i>lo</i> or <b>loopback</b> <i>lo<math>n</math></i>	<i>n</i> : The name of a loopback interface, where <i>n</i> ranges from 1 through 99999.
OpenVPN	<b>openvpn</b> <i>vtunx</i>	<i>vtunx</i> : The identifier of an OpenVPN interface. The identifier ranges from vtun0 through vtun $x$ , where <i>x</i> is a nonnegative integer.
Tunnel	<b>tunnel</b> <i>tunx</i> or <b>tunnel</b> <i>tunx</i> <b>parameters</b>	<i>tunx</i> : The identifier of a tunnel interface you are defining. The identifier ranges from tun0 through tun $x$ , where <i>x</i> is a nonnegative integer.

Interface Type	Syntax	Parameters
Virtual tunnel	<b>vti</b> <i>vtix</i>	<p><i>vtix</i>: The identifier of a virtual tunnel interface you are defining. The identifier ranges from vti0 through vti<i>x</i>, where <i>x</i> is a nonnegative integer.</p> <p><b>Note:</b> This interface does not support IPv6.</p>
VRRP	<i>parent-interface</i> <b>vrrp</b> <b>vrrp-group</b> <i>group</i>	<p><i>parent-interface</i>: The type and identifier of a parent interface; for example, data plane dpOp1p2 or bridge br999.</p> <p><i>group</i>: A VRRP group identifier.</p> <p>The name of a VRRP interface is not specified. The system internally constructs the interface name from the parent interface identifier plus the VRRP group number; for example, dpOp1p2v99. Note that VRRP interfaces support the same feature set as does the parent interface.</p>

# List of Acronyms

---

ACL	access control list
ADSL	Asymmetric Digital Subscriber Line
API	Application Programming Interface
AS	autonomous system
ARP	Address Resolution Protocol
BGP	Border Gateway Protocol
BIOS	Basic Input Output System
BPDU	Bridge Protocol Data Unit
CA	certificate authority
CHAP	Challenge Handshake Authentication Protocol
CLI	command-line interface
DAD	Duplicate Address Detection
DDNS	dynamic DNS
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol version 6
DLCI	data-link connection identifier
DMI	desktop management interface
DMZ	demilitarized zone
DN	distinguished name
DNS	Domain Name System
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
eBGP	external BGP
EGP	Exterior Gateway Protocol
ECMP	equal-cost multipath
ESP	Encapsulating Security Payload
EUI-64	IEEE-defined 64-bit extended unique identifier
FIB	Forwarding Information Base
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HDLC	High-Level Data Link Control
I/O	Input/Output
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IGP	Interior Gateway Protocol
IPS	Intrusion Protection System
IKE	Internet Key Exchange
IP	Internet Protocol

IPOA	IP over ATM
IPsec	IP security
IPv4	IP Version 4
IPv6	IP Version 6
IS-IS	Intermediate System-to-Intermediate System
ISP	Internet Service Provider
L2TP	Layer 2 Tunneling Protocol
LACP	Link Aggregation Control Protocol
LAN	local area network
LDAP	Lightweight Directory Access Protocol
LLDP	Link Layer Discovery Protocol
MAC	medium access control
MIB	Management Information Base
MLPPP	multilink PPP
MPLS	Multiprotocol Label Switching
MPLS EXP	MPLS experimental
MPLS TE	MPLS Traffic Engineering
MRRU	maximum received reconstructed unit
MTU	maximum transmission unit
NAT	Network Address Translation
ND	Neighbor Discovery
NIC	network interface card
NS	Neighbor Solicitation
NTP	Network Time Protocol
OSPF	Open Shortest Path First
OSPFv2	OSPF Version 2
OSPFv3	OSPF Version 3
PAM	Pluggable Authentication Module
PAP	Password Authentication Protocol
PAT	Port Address Translation
PCI	peripheral component interconnect
PKI	Public Key Infrastructure
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PVC	permanent virtual circuit
QoS	quality of service
RADIUS	Remote Authentication Dial-In User Service
RA	router advertisement
RIB	Routing Information Base
RIP	Routing Information Protocol
RIPng	RIP next generation



RS	router solicitation
Rx	receive
SLAAC	Stateless address auto-configuration
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
SONET	Synchronous Optical Network
SSH	Secure Shell
STP	Spanning Tree Protocol
TACACS+	Terminal Access Controller Access Control System Plus
TCP	Transmission Control Protocol
ToS	Type of Service
Tx	transmit
UDP	User Datagram Protocol
vif	virtual interface or VLAN interface
VLAN	virtual LAN
VPN	Virtual Private Network
VRRP	Virtual Router Redundancy Protocol
WAN	wide area network